

System Automation for z/OS



User's Guide

Version 2 Release 3

System Automation for z/OS



User's Guide

Version 2 Release 3

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page xi.

Eleventh Edition (October 2005)

This edition applies to System Automation for z/OS (Program Number 5645-006) Version 2, Release 3, an IBM licensed program, and to all subsequent releases and modifications until otherwise indicated in new editions or technical newsletters.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

IBM welcomes your comments. A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

IBM Deutschland Entwicklung GmbH
Department 3248
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany

If you prefer to send comments electronically, use one of the following methods:

FAX (Germany): 07031 + 16-3456
FAX (Other Countries): (+49)+7031-16-3456
Internet: s390id@de.ibm.com

Internet

Visit our homepage at <http://www.ibm.com/servers/eserver/zseries/software/sa>

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1996, 2004. All rights reserved.

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

Contents

Figures vii

Tables ix

Notices xi

Web Site Disclaimer xi
Programming Interface Information xi
Trademarks xii

About This Book xiii

Who Should Use This Book. xiii
Where to Find More Information xiii
 The System Automation for z/OS Library xiii
 Related Product Information xiv
 Using LookAt to look up message explanations xvi
 Accessing z/OS licensed documents on the
 Internet xvi
Related Online Information xvi

Part 1. Introduction and Concepts . . 1

Chapter 1. Introducing SA z/OS 3

Overview 3
Component Description. 4
 System Operations 4
 Processor Operations 4
 I/O Operations 5
Hardware Overview 5
 SA z/OS Resources 6
 Channels 9
 Processors, Systems. 10
 I/O Devices 11
Automation Policy 11

Chapter 2. Concepts 13

Automation Concepts 13
 How Requests Become Goals 19
 How Request Propagation Works 19
 Conflicting Relationships and Requests 20
 How the Automation Manager and Automation
 Agents Cooperate in a Sysplex 21
Controlling Application Move and Server
Management 22
 Move Functions 22
 Controlling Application Move with Preference
 Values 23
SA z/OS Automation Flags 31
 Automation Agent Flags 31
 The Automation Manager Global Automation
 Flag 32
 The Hold Flag 33
Statuses Supplied by the Automation Manager 33
 Observed Status 34
 Automation Status 35

 Desired Status 36
 Startability Status 36
 Compound Status 37
 Health Status 37
Statuses Supplied by Automation Agents 38
 Automation Agent Statuses 38
 Application Monitor Status 45
 Automatic Restart Manager Statuses 46
 Mapping the Automation Agent Statuses to the
 Automation Manager Observed Status 47
 SA z/OS Processes Involved in Status
 Determination 48
Using SA z/OS for Monitoring. 50
 Resynchronization 50
Using Group Start and Stop Types. 51
 Basic Groups 51
 Move and Server Groups 52

Chapter 3. Starting and Stopping

SA z/OS 53

Starting, Stopping, and Maintaining the Automation
Manager 53
 Starting the Automation Manager—START (S). 53
 Stopping the Automation Manager—STOP (P). 54
 Making an Automation Manager the Primary
 One 54
 Stopping the Automation Manager with Different
 Options. 54
 Getting Automation Manager Diagnostic
 Information 55
 Takeover File Repair 55
Starting NetView (System Operations) 57
Starting and Stopping I/O Operations 57
Starting and Stopping Processor Operations 58
Starting and Stopping Applications 58

Chapter 4. Refreshing Automation

Policy 61

Refreshing an Automation Manager Configuration 62
Identifying the Current Automation Control File and
NetView Automation Table 62
Identifying the Current AMC File Used by the
Automation Manager 63

Part 2. Using SA z/OS on the Host 65

Chapter 5. Getting Started 67

Issuing Commands 67
Using Command Dialogs. 68
 How to Get Help 69

**Chapter 6. How to Get Resource
Information Sysplex-Wide 71**

Commands that Are Processed by the Automation Manager	71
How to Use INGLIST to View Resources	71
How to Use INGINFO to View a Selected Resource	74
How to Display Relationship Information: INGRELS	76
Commands that Are Processed by the Automation Agents	76
How to Display the Automation Agent View with DISPSTAT	77
How to Use DISPINFO to Display Detailed Information	81
How to Use DISPTREE to Display Dependency Information	84
How to Display Statistical Information	85

Chapter 7. How to Monitor and Control Resources 89

How to Start and Stop Resources	89
How to Use INGREQ	89
How to Use INGVOTE	95
How to Use INGHIST	96
How to Use Schedules—INGSCHED	97
How to Display Trigger Information—INGTRIG and DISPTRG	99
How to Display Event Information—DISPEVT, DISPEVTS, and INGEVENT	101
Determining Automation Agent Flag Values	103
How to Work with Application Groups.	107
How to Set a Timer	112
How to Interact with the Automation Manager - INGAMS	113
How to Change the Automation Agent Status - SETSTATE	114
How to Monitor Messages	117
How to Monitor Hardware	122
How to Monitor Health Performance	124
How to Monitor Processor Status.	126
Monitoring with the Status Display Facility (SDF)	133
SDF Focal Point Monitoring	134
Interpreting Colors in the SDF and DISPSTAT	135
Using SDF	136
How SDF Works	137
How to Monitor WTORs with SDF	140
How to Monitor Application Groups with SDF	140
How to Monitor and Control a Sysplex.	141
How to Work with INGPLEX	141
How to Work with INGHC.	175
How SA z/OS Uses Error Thresholds	179
Displaying Threshold Settings and Occurrences	180
How to Set or Change Error Thresholds	183
Handling Jobs that are not Controlled by SA z/OS	183
Communicating with Other Systems.	184
Sending Commands within a Sysplex	184
Sending Commands to a Linux Target System	187
Terminal Access Facility Full-Screen Sessions	188
Gateway Sessions	190

Chapter 8. Scenario: Finding Out Why a Resource Is not Available 191

Chapter 9. Using I/O Operations 201

Making Connectivity Changes.	202
Display Commands To Obtain Data About I/O Resources	202
Connectivity Commands That Affect a Switch	202
Display and Utility Commands Related to Connectivity Commands	203
Utility Commands for General Processing.	203
Using the I/O Operations ISPF Dialog	204
Using the z/OS Operator Console for I/O Operations Commands	204

Chapter 10. VM Second Level Systems Support. 207

ProcOps Dialog.	207
Status Summary Panel (ISQESUM)	208
Target Hardware Summary Panel (ISQEHARP)	208
Target System Summary Panel (ISQETARG)	210
Guests Summary Panel (ISQETRGG)	210
PATH Detail Panel (ISQEPNVC)	211
PSM Console Mode	211
Starting the PSM Control Program	212
Stopping the PSM Control Program	213
Restarting the PSM Control Program	214

Part 3. Using SA z/OS on Workstations 215

Chapter 11. Using the NetView Management Console for SA z/OS . . . 217

Introduction.	217
Starting the NMC	218
Monitoring SA z/OS on the NMC	219
Regular Resources.	219
Minor Resources, Anchors and Tokens	220
Properties of SA z/OS Objects on the NMC Workstation	223
Dynamic Behavior of the SA z/OS Topology Manager	225
SA z/OS Topology Manager Initialization: INGTPO INIT	225
System Detection	225
The Heartbeat Function	225
Takeover when a System Goes Down	226
Issuing Commands from the NMC	226
Direct Commands	226
Dialog Commands	226
Interactive Commands	227
The INGRTCMD Command	229
Creating Views	230
Network Views and Exception Views	230
Names	231
SA z/OS Topology Manager Commands	233
Processor Operations on the NMC	234
Monitoring Sysplex Resources.	235
Problem Management	235

Exception Views	235	C. If You Select the Options Menu	270
Marked and Unmarked Resources	236	D. If You Select the Help Menu	271
Updating Operator Data.	236		
Problem Management of Minor Resources.	237		
<hr/>			
Part 4. TEC Notification by SA z/OS	239	Appendix C. Problem Determination	273
Chapter 12. Using TEC Notification by SA z/OS	241	Concurrent User Access within the Customization Dialog.	273
Understanding the Use of TEC Notification by SA z/OS.	241	Maintaining System Status during a Status Forwarding Path Failure.	273
Introduction of TEC Notification by SA z/OS	242	Debugging and Tracing	274
SA z/OS Messages, Alerts and Tivoli Events.	242	Using Trace Services for the Automation Manager	274
Mapping SA z/OS Messages to Tivoli Events	243	Using AOCTRACE	276
Mapping NetView Alerts to Tivoli Events	249	How to Use a Log with INGHIST	276
TEC Rules for SA z/OS Events	249	How to Use the Diagnostic Option of INGAMS	277
Operation of TEC Notification by SA z/OS	251	Tracing and Debugging for ProcOps and the BCP Internal Interface	278
Startup of TEC Notification by SA z/OS	251		
How TEC Notification by SA z/OS Works	253	Appendix D. Additional BCP Internal Interfaces	281
Termination of TEC Notification by SA z/OS	256	Determine BCP Internal Interface Communication Task Names	281
<hr/>			
Part 5. Appendixes	257	BCP Internal Interface Recovery Actions	281
Appendix A. Automation Initialization Options	259	Recovery Case 1: <tn>_task_module_INGHWCOM_not_running	281
Appendix B. Using the I/O Operations ISPF Dialog	263	Recovery Case 2: <tn>_reached QueueLimit_<ql>	282
Getting Started	263	Recovery Case 3: <tn>_task_is_not_available	282
Starting with the I/O Operations ISPF Main Menu	264	Terminating NetView and the BCP Internal Interface	283
A. If You Select the Modify Menu	265		
A Sample Way to Modify a Switch Matrix.	266	Appendix E. The IBM Health Checker for z/OS and Sysplex Checks	285
B. If You Select the View Menu	267		
		Glossary	299
		Index	319

Figures

1. A Basic Hardware Configuration	5	52. Target System Summary Panel	127
2. Conflicting Relationships and Requests	20	53. Path Detail Panel	131
3. Automation Manager and Automation Agents Cooperation in a Sysplex	22	54. Example SDF System Summary Panel	137
4. Non-Disruptive and Disruptive Restoration	28	55. Example SDF Hierarchy for KEY1.	138
5. Observed Status Transitions	35	56. Example SDF Detail Status Display Panel for HUGC2APL0	139
6. Automations Status Transitions	36	57. SDF Summary Status Panel	140
7. Application Startup Under SA z/OS's Control	59	58. SDF Detail Status Display Panel for an Application Group	141
8. Application Stop Under SA z/OS's Control	60	59. INGPLEX Selection Panel	142
9. Command Dialog DISPSYS	63	60. INGPLEX SYStem Panel	143
10. Command Dialog INGAMS	64	61. INGPLEX Console Panel	144
11. System Operations Main Menu	68	62. INGCF Panel	146
12. SA z/OS Command Dialogs Main Menu	69	63. DRAIN CF Command Dialog	148
13. INGLIST Command Dialog	72	64. DRAIN Command Dialog: Confirmation Panel for REBUILD	149
14. INGFLT Command Dialog	73	65. DRAIN Command Dialog: Rebuild Process	149
15. INGINFO Command Dialog	75	66. DRAIN Command Dialog: Panel after Rebuild	150
16. INGRELS Command Dialog	76	67. DRAIN Command Dialog: Panel after Forcing	150
17. DISPSTAT Command Dialog	77	68. DRAIN Command Dialog: Panel after Draining	151
18. DISPFILT Command Dialog	80	69. DRAIN Command Dialog: Panel after Inactivation	151
19. DISPINFO Command Dialog	81	70. ENABLE Command Dialog: Panel before any Action	153
20. Statistics Summary — INGRPT Command Dialog	86	71. Confirmation Panel for ENABLE	154
21. Statistical Information about Resources — INGRPT Command Dialog	87	72. ENABLE Command Dialog: Panel after Enabling	154
22. Detailed Statistical Resource Information	88	73. ENABLE Command Dialog: Panel after Populating	155
23. INGREQ Command Dialog	90	74. Display Sender Path Panel	155
24. INGREQ Command Dialog	92	75. Display Structures Panel	156
25. INGREQ Verification	93	76. INGPLEX CDS Panel	159
26. INGVOTE Command Dialog	95	77. Confirmation Panel for Switching from the Current Primary CDS to the Alternate CDS	161
27. Expanded Workitem	97	78. INGPLEX CDS Dialog Panel after the Switch	162
28. INGSCHED Command Dialog	97	79. CFRM Couple Data Set Information Panel before Policy Switch	162
29. INGSCHED Details	98	80. Confirmation Panel for Policy Switch	163
30. Panel for Schedule Overrides	98	81. Channel Path Information for CFRM Couple Data Sets	164
31. Panel for Resource Overrides	99	82. INGPLEX IPL Main Panel	165
32. Trigger Condition List	101	83. INGPLEX DUMP Panel	166
33. DISPEVT Command Dialog	102	84. INGPLEX SDUMP Panel	167
34. DISPEVTS Command Dialog	102	85. INGPLEX SDUMP Modification Panel	168
35. INGEVENT Command Dialog	103	86. INGPLEX SVCDUMP Target System Selection Panel	169
36. Example Display of Status of all Automated Resources (DISPSTAT)	104	87. INGPLEX SVCDUMP Address Space Selection Panel	169
37. DISPFLGS Command Dialog Panel	105	88. INGPLEX SVCDUMP Address Space Detail Panel	170
38. INGAUTO Command Dialog	107	89. INGPLEX SVCDUMP Dump Option Panel	171
39. INGGROUP Command Dialog (1).	110	90. INGPLEX SLIP Main Panel	172
40. INGGROUP Command Dialog (2).	111	91. INGPLEX BESTpractices Panel	173
41. SETTIMER Command Dialog	112	92. INGHC Panel	176
42. Add/Change Timer	113		
43. INGAMS Command Dialog	114		
44. Transitions for a Normal Subsystem	115		
45. Transitions for an Abending Subsystem	116		
46. SETSTATE Command Dialog	117		
47. ISQXDST Command on NetView Console	123		
48. Target Hardware Summary Panel for OCF-Based Processors	123		
49. DISPMTR Command Dialog Panel	125		
50. DISPMTR Command Dialog Details Panel	125		
51. ProcOps Target Status Summary Panel	126		

93. INGHC Check Confirmation Panel	178	121. Target Hardware Summary Panel for Non-PSMs ISQEHARP	209
94. INGHC Line Mode Output	179	122. Target System Summary Panel ISQETARG	210
95. INGTHRES Command Dialog	181	123. PSMs/Guests Summary Panel ISQETRGG	210
96. DISPERRS Command Dialogs Panel	182	124. PATH Detail Panel ISQEPNVC.	211
97. DISPASF Command Response	183	125. Overview	218
98. INGLKUP Command Dialog Panel	184	126. Tivoli NetView for z/OS Management Console	219
99. Command Domain Menu Panel	187	127. Network View for Regular Resources	220
100. Sending a Command to a Linux Target System.	188	128. Subview with Applications of an Application Group	220
I 101. TAF Fullscreen Display Panel	189	129. Anchor View.	222
102. INGLIST Output	191	130. View with 'Lost&Found'-Anchors.	223
103. INGVOTE Output	192	131. Resource Properties Window	224
104. Refreshed INGVOTE Output	193	132. Parameter Window for INGTHRES	227
105. INGINFO Output	193	133. NetView Console with INGREQ Panel	228
106. Current Automation Situation	194	134. Actualized View	229
107. INGLIST Output for MSCSK* Resources	194	135. Processor Operations View	234
108. INGVOTE Output for MSCSK2	195	136. Exception View	236
109. INGSCHED for MSCSK2.	195	137. Tivoli Event Inheritance	246
110. Schedule Override for MSCSK2	196	138. Tivoli desktop with started Tivoli Enterprise Console	253
111. Refreshed INGLIST Output	196	139. Invoking Predefined Actions for Events	254
112. Checking Automation Flags with INGLIST	197	140. TEC Tasks Dialog	255
113. Checking Triggers with INGLIST	197	141. NetView 3270 Management Console	256
114. Complete Automation Environment for MSCSK1	198	142. Sample AOF767I Message	261
115. DISPTRG Command Dialog.	198	143. INGHIST Logger Messages	276
116. Trigger Condition List	199	144. Diagnostic	277
117. Verifying Automation Status	199	145. INGAMS Command Dialog, Show Details	278
118. ProcOps Panel Flow	207		
119. Status Summary Panel ISQESUM	208		
120. Target Hardware Summary Panel for PSMs ISQEHARP	208		

Tables

1.	System Automation for z/OS Library	xiii	9.	Table of Default Subsystem Status Colors and Priorities	135
2.	Related Products Books	xiv	10.	Mapping SA z/OS Messages to SA z/OS Events	243
3.	Hierarchy of Requests	15	11.	SA z/OS Events and Corresponding Mapped SA z/OS Messages	246
4.	Preference Values and Behavior of Members	25	12.	Mapping SA z/OS Alerts to SA z/OS Events	249
5.	Preference Bonuses	26	13.	Overview of HealthChecker Best Practices Checks	285
6.	Regular Monitoring: Automation Status Determination	46			
7.	Automation Agent to Automation Manager Status Mapping	47			
8.	SA z/OS Message Classes	119			

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter 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
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Deutschland Entwicklung GmbH
Department 3248
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany

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

Web Site Disclaimer

Any pointers in this publication to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement. IBM accepts no responsibility for the content or use of non-IBM Web sites specifically mentioned in this publication or accessed through an IBM Web site that is mentioned in this publication.

Programming Interface Information

This publication documents information that is NOT intended to be used as a Programming Interface of System Automation for z/OS.

Trademarks

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

CICS	DB2	ESCON
IBM	IMS	MVS
NetView	OS/390	PR/SM
PS/2	RACF	RMF
Sysplex Timer	Tivoli	Tivoli Enterprise Console
VTAM	z/OS	z/VM
zSeries		

The following terms are trademarks of other companies:

- Linux is a registered trademark of Linus Torvalds.

About This Book

This book describes System Automation for z/OS[®] (SA z/OS) and contains information on how to use SA z/OS to monitor and control your systems.

This document provides an overview of the operator tasks. It points you to ways to manage your systems. For further detail, refer to the help panels or online documentation available on both 3270 panels and workstation windows.

Who Should Use This Book

This book is intended primarily for operators, but may also be useful for others. The information provided is specifically about using SA z/OS and assumes the following knowledge and experience:

- An understanding of z/OS
- An understanding of basic network concepts and terminology
- Using NetView[®]
- Using the NetView Management Console (NMC)
- Responding to messages and some troubleshooting

System administrators, programmers and help desk personnel may find the information helpful for installation, maintenance, and investigating and correcting problems. It also helps educate users about the SA z/OS functions.

Where to Find More Information

The System Automation for z/OS Library

The following table shows the information units in the System Automation for z/OS library:

Table 1. System Automation for z/OS Library

Title	Order Number
<i>System Automation for z/OS Planning and Installation</i>	SC33-8260
<i>System Automation for z/OS Customizing and Programming</i>	SC33-8261
<i>System Automation for z/OS Defining Automation Policy</i>	SC33-8262
<i>System Automation for z/OS User's Guide</i>	SC33-8263
<i>System Automation for z/OS Messages and Codes</i>	SC33-8264
<i>System Automation for z/OS Operator's Commands</i>	SC33-8265
<i>System Automation for z/OS Programmer's Reference</i>	SC33-8266
<i>System Automation for z/OS CICS Automation Programmer's Reference and Operator's Guide</i>	SC33-8267
<i>System Automation for z/OS IMS Automation Programmer's Reference and Operator's Guide</i>	SC33-8268
<i>System Automation for z/OS OPC Automation Programmer's Reference and Operator's Guide</i>	SC23-8269
<i>System Automation for z/OS Licensed Program Specifications</i>	GI11-2690

The System Automation for z/OS books are also available on CD-ROM as part of the following collection kit:

IBM® Online Library z/OS Software Products Collection (SK3T-4270)

SA z/OS Home Page

For the latest news on SA z/OS, visit the SA z/OS home page at <http://www.ibm.com/servers/eserver/zseries/software/sa>

Related Product Information

The following table shows the books in the related product libraries that you may find useful for support of the SA z/OS base program.

Table 2. Related Products Books

Title	Order Number
<i>ISPF User's Guide</i>	SC34-4484
<i>ISPF Dialog Management Guide and Reference</i>	SC34-4266
<i>MVS/ESA™ MVS Configuration Program Guide and Reference</i>	GC28-1817
<i>MVS/ESA Planning: Dynamic I/O Configuration</i>	GC28-1674
<i>MVS/ESA Support for the Enterprise Systems Connection</i>	GC28-1140
<i>MVS/ESA Planning: APPC Management</i>	GC28-1110
<i>MVS/ESA Application Development Macro Reference</i>	GC28-1822
<i>OS/390®: MVS System Commands</i>	GC28-1781
<i>MVS/ESA SPL Application Development Macro Reference</i>	GC28-1857
<i>OS/390 Hardware Configuration Definition: User's Guide</i>	SC28-1848
<i>OS/390 Information Roadmap</i>	GC28-1727
<i>OS/390 Information Transformation</i>	GC28-1985
<i>OS/390 Introduction and Release Guide</i>	GC28-1725
<i>OS/390 JES Commands Summary</i>	GX22-0041
<i>OS/390 Licensed Program Specifications</i>	GC28-1728
<i>OS/390 Printing Softcopy Books</i>	S544-5354
<i>OS/390 Starting Up a Sysplex</i>	GC28-1779
<i>OS/390 Up and Running!</i>	GC28-1726
<i>Planning for the 9032 Model 3 and 9033 Enterprise Systems Connection Director</i>	SA26-6100
<i>Resource Access Control Facility (RACF®) Command Language Reference</i>	SC28-0733
<i>S/390® MVS Sysplex Overview -- An Introduction to Data Sharing and Parallelism</i>	GC23-1208
<i>S/390 MVS Sysplex Systems Management</i>	GC23-1209
<i>S/390 Sysplex Hardware and Software Migration</i>	GC23-1210
<i>S/390 MVS Sysplex Application Migration</i>	GC23-1211
<i>S/390 Managing Your Processors</i>	GC38-0452
<i>Tivoli/Enterprise Console User's Guide Volume I</i>	GC31-8334
<i>Tivoli/Enterprise Console User's Guide Volume II</i>	GC31-8335

Table 2. Related Products Books (continued)

Title	Order Number
<i>Tivoli/Enterprise Console Event Integration Facility Guide</i>	GC31-8337
<i>Tivoli® NetView for OS/390 Administration Reference</i>	SC31-8222
<i>Tivoli NetView for OS/390 Application Programming Guide</i>	SC31-8223
<i>Tivoli NetView for OS/390 APPN Topology and Accounting Agent</i>	SC31-8224
<i>Tivoli NetView for OS/390 Automation Guide</i>	SC31-8225
<i>Tivoli NetView for OS/390 AON Customization Guide</i>	SC31-8662
<i>Tivoli NetView for OS/390 AON User's Guide</i>	GC31-8661
<i>Tivoli NetView for OS/390 Bridge Implementation</i>	SC31-8238
<i>Tivoli NetView for OS/390 Command Reference Vol. 1</i>	SC31-8227
<i>Tivoli NetView for OS/390 Command Reference Vol. 2</i>	SC31-8735
<i>Tivoli NetView for OS/390 Customization Guide</i>	SC31-8228
<i>Tivoli NetView for OS/390 Customization: Using Assembler</i>	SC31-8229
<i>Tivoli NetView for OS/390 Customization: Using Pipes</i>	SC31-8248
<i>Tivoli NetView for OS/390 Customization: Using PL/I and C</i>	SC31-8230
<i>Tivoli NetView for OS/390 Customization: Using REXX and CLIST Language</i>	SC31-8231
<i>Tivoli NetView for OS/390 Data Mode Reference</i>	SC31-8232
<i>Tivoli NetView for OS/390 Installation: Getting Started</i>	SC31-8767
<i>Tivoli NetView for OS/390 Installation: Migration Guide</i>	SC31-8768
<i>Tivoli NetView for OS/390 Installation: Configuring Graphical Components</i>	SC31-8770
<i>Tivoli NetView for OS/390 Installation: Configuring Additional Components</i>	SC31-8769
<i>Tivoli NetView for OS/390 Messages and Codes</i>	SC31-8237
<i>Tivoli NetView for OS/390 MultiSystem Manager User's Guide</i>	SC31-8607
<i>Tivoli NetView for OS/390 NetView Management Console User's Guide</i>	GC31-8665
<i>Tivoli NetView for OS/390 User's Guide</i>	SC31-8241
<i>Tivoli NetView for OS/390 RODM and GMFHS Programming Guide</i>	SC31-8233
<i>Tivoli NetView for OS/390 Security Reference</i>	SC31-8606
<i>Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide</i>	SC31-8239
<i>Tivoli Management Platform Reference Guide</i>	GC31-8324
<i>TSO/E REXX/MVS User's Guide</i>	SC28-1882
<i>TSO/E REXX/MVS Reference</i>	SC28-1883
<i>VM/XA SP™ GCS Command and Macro Reference</i>	SC23-0433
<i>VSE/SP Unattended Node Support</i>	SC33-6412
<i>VTAM® Messages and Codes</i>	SC31-6493
<i>VTAM V3R3 Network Implementation Guide</i>	SC31-6404
<i>VTAM V3R4 Network Implementation Guide</i>	SC31-6434

Using LookAt to look up message explanations

LookAt is an online facility that lets you look up explanations for most messages you encounter, as well as for some system abends and codes. Using LookAt to find information is faster than a conventional search because in most cases LookAt goes directly to the message explanation.

You can access LookAt from the Internet at:

<http://www.ibm.com/eserver/zseries/zos/bkserv/lookat/> or from anywhere in z/OS or z/OS.e where you can access a TSO/E command line (for example, TSO/E prompt, ISPF, z/OS UNIX System Services running OMVS).

The LookAt Web site also features a mobile edition of LookAt for devices such as Pocket PCs, Palm OS, or Linux-based handhelds. So, if you have a handheld device with wireless access and an Internet browser, you can now access LookAt message information from almost anywhere.

To use LookAt as a TSO/E command, you must have LookAt installed on your host system. You can obtain the LookAt code for TSO/E from a disk on your z/OS *Collection* (SK3T-4269) or from the LookAt Web site's **Download** link.

Accessing z/OS licensed documents on the Internet

z/OS licensed documentation is available on the Internet in PDF format at the IBM Resource Link™ Web site at:

<http://www.ibm.com/servers/resourceLink>

Licensed documents are available only to customers with a z/OS license. Access to these documents requires an IBM Resource Link user ID and password, and a key code. With your z/OS order you received a Memo to Licensees, (GI10-0671), that includes this key code.¹

To obtain your IBM Resource Link user ID and password, log on to:

<http://www.ibm.com/servers/resourceLink>

To register for access to the z/OS licensed documents:

1. Sign in to Resource Link using your Resource Link user ID and password.
2. Select **User Profiles** located on the left-hand navigation bar.

Note: You cannot access the z/OS licensed documents unless you have registered for access to them and received an e-mail confirmation informing you that your request has been processed.

Printed licensed documents are not available from IBM.

You can use the PDF format on either **z/OS Licensed Product Library CD-ROM** or IBM Resource Link to print licensed documents.

Related Online Information

SA z/OS provides online help for all parts of its operator interfaces. From any terminal display, press PF1 to obtain help for that display.

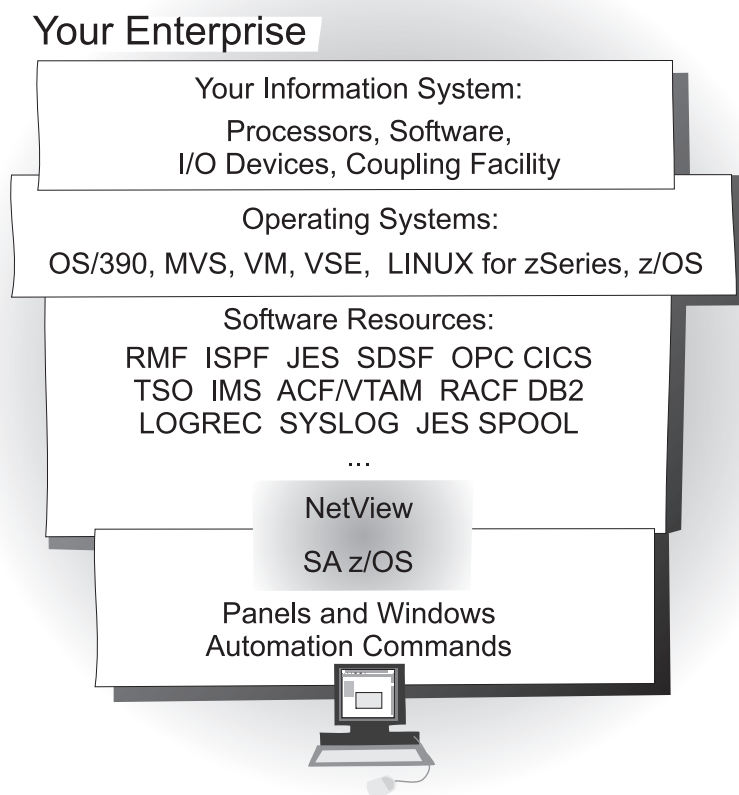
1. z/OS.e™ customers received a Memo to Licensees, (GI10-0684) that includes this key code.

SA z/OS also provides online help panels to support the tasks of installation and customization.

Part 1. Introduction and Concepts

Chapter 1. Introducing SA z/OS

System Automation for z/OS (SA z/OS) is a NetView-based application designed to provide a single point of control for a full range of systems management functions. SA z/OS plays a key role in supplying high-end automation solutions. SA z/OS functions include monitoring, controlling and automating a large range of system elements spanning both the hardware and software resources of your enterprise.



Overview

SA z/OS is a systems management program with a single point of control. You see a single system image for a full range of essential systems management functions.

Monitor

Monitor your resources to respond before they affect end users:

- Monitor hardware components
- Monitor software products and applications
- Monitor automated processes
- Monitor messages and alerts

Control

Take action to control conditions:

Overview

- Start and stop your entire enterprise system; initiate hardware and software startup and shutdown sequences
- Manage both remote and local operations and support any zSeries® and 390-CMOS processor within a Parallel Sysplex
- Manage several operating systems: z/OS, OS/390, MVS™, VM, VSE, and Linux™ for zSeries
- Control a coupling facility as a target system with coupling links in a Parallel Sysplex environment
- React to errors and unscheduled events

Automate

Automate many repetitive and complex tasks:

- Start/shutdown software resources
- Start/shutdown hardware resources
- Control channels and channel paths
- Control availability of I/O devices
- Control switching of I/O device ports
- Detect and respond to system messages
- Perform initial program load (IPL)
- Perform system power-on reset (POR)
- Build automation policy for your enterprise
- Extend the built-in automation routines by writing your own automation policies

You monitor and control hardware and software resources from a NetView Management Console (NMC) workstation or from a NetView console.

Component Description

System Operations

System operations monitors and controls system operations applications and subsystems such as NetView, SDSF, JES, RMF™, TSO, RODM, ACF/VTAM, DB2®, CICS®, IMS™, and OPC.

With system operations, you can automate Parallel Sysplex applications. SA z/OS can automate applications distributed over a sysplex by virtually removing system boundaries for automation through its automation manager/automation agent design. SA z/OS reduces the complexity of managing a Parallel Sysplex through its goal driven automation and its concepts such as grouping and powerful dependency support which enable you to model your configuration. Single systems are also fully supported; the automation scope is then just one system.

Enterprise monitoring is used by SA z/OS to update the NetView Management Console (NMC) resource status information which is stored in the Resource Object Data Manager (RODM).

Processor Operations

Processor operations monitors and controls processor hardware operations. It provides a connection from a focal point processor to a target processor. With NetView on the focal point system, processor operations automates operator and system consoles for monitoring and recovering target processors.

Processor operations allows you to power on and off multiple target processors and reset them, perform IPLs, set the time of day clocks, respond to messages, monitor status, and detect and resolve wait states.

I/O Operations

I/O operations provides a single point of control for managing connectivity in your active I/O configurations. It takes an active role in detecting unusual I/O conditions and lets you view and change paths between a processor and an input/output device, which can involve using dynamic switching (the ESCON[®] switch).

I/O operations changes paths by letting you control channels, ports, switches, control units, and input/output devices. You can do this through an operator console or API.

Hardware Overview

Figure 1 illustrates a basic hardware configuration.

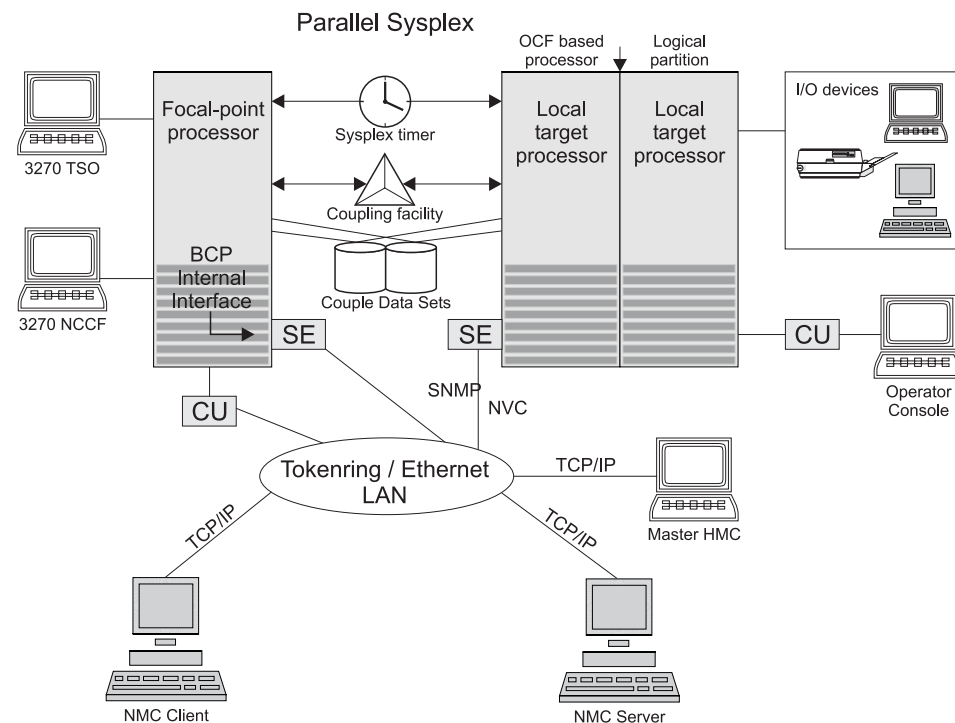


Figure 1. A Basic Hardware Configuration

It shows a two-processor Parallel Sysplex configuration with systems running on it. One is playing the role of a SA z/OS focal point. For example, the role of the SA z/OS NMC focal point with information about all the systems and applications in the sysplex, running under the control of SA z/OS.

Operators can use a workstation with the SA z/OS NMC client code installed, to work with graphical views of the SA z/OS controlled resources stored on the focal point. The NMC server component receives status changes from the NMC focal point and distributes them to the registered clients to update their dynamic resource views. Sysplex specific facilities, like the coupling facility hardware can be

Hardware Overview

managed and controlled using the NMC's client graphical interface, as well as the 3270 NCCF based SA z/OS operator interfaces.

With the same interfaces, processor operations, another SA z/OS focal point function can be operated. With processor operations it is possible to manage and control the complete processor hardware in a sysplex. Operator tasks such as re-IPLing a sysplex member or activating a changed processor configuration can be accomplished. Processor operations uses the processor hardware infrastructure, consisting of the CPC Support Element (SE), or the Hardware Management Console (HMC) interconnected in a processor hardware LAN, to communicate with the own, other local, or remote located Support Elements of other CPCs. The Support Elements provide the Systems Management Interface OCF (Operations Command Facility) to perform hardware commands like LOAD or SYSTEM RESET to control the hardware and hardware images. SA z/OS processor operations can be customized to use SNA-based NetView connections (NVC), or IP based SNMP for communication. For Parallel Sysplex environments, SA z/OS provides an additional processor hardware interface, the BCP (basic control program) internal interface. This interface is independent from processor operations. It allows processor hardware operation in a sysplex, without requiring external network CUs (control units). From a system in the sysplex, the SE of the own CPC as well as the SEs of the other processors in the sysplex can be accessed.

SA z/OS Resources

The following describes the functions and resources used by SA z/OS and its components.

Focal point processor	In a multisystem environment, the primary (or <i>issuing</i>) host processor has SA z/OS installed. You might think of it as SA z/OS's <i>command center</i> . This is where you enter commands you want to process. It includes a central processor complex which is a collection of hardware that consists of central storage, one or more central processors, timers, and channels. NetView and SA z/OS are installed on this processor.
Target processor	The processor that is controlled by a focal point processor. It can be a local or remote processor, or a logical partition of a processor. When you send a command from a focal point system, the target is the processor you want the command to affect. The operating system can be z/OS, OS/390, VM, VSE, or a coupling facility. If the operating system is z/OS, SA z/OS should be installed on the target processor for full SA z/OS functionality. If the operating system is VM, or VSE, or if the target processor is a coupling facility, only SA z/OS processor operations functions are available for the target.
OCF-based processor	CMOS-S/390, zSeries operations command facility (OCF) for interacting with human operators or external programs to perform operations management functions on the central processor. Processor operations uses this facility to manage and control the processors.

Parallel Sysplex	<p>A set of z/OS systems communicating and cooperating with each other through certain multisystem hardware components (coupling devices and sysplex timers) and software services (couple data sets). In a Parallel Sysplex, z/OS provides the coupling services that handle the messages, data, and status for the parts of a multisystem application that has its workload spread across two or more of the connected processors. Sysplex timers, coupling facilities, and couple data sets containing policy and states for basic simplex functions are all parts of a sysplex.</p> <p>An z/OS sysplex can be viewed and managed as a single system image on the NMC workstation.</p>
Logically Partitioned (LPAR) mode	<p>A processor with the PR/SM™ feature that can be divided into partitions with separate logical system consoles that allocates hardware resources among several logical partitions. (It is called <i>logical</i> because the processor is not physically divided, but divided only by definition.) The partitions are defined, monitored, and activated separately by processor operations.</p> <p>A processor that does not use logical partitions is in “basic mode”.</p>
Images	<p>Processors can operate as <i>single images</i> or <i>partitioned images</i>. A single image allows a multiprocessor system to function as one central processor complex.</p> <p>A multiprocessor system that functions as two or more independent processors (with separate power, water, and maintenance boundaries) is a <i>physically partitioned configuration</i> image.</p>
Communications links	<p>Paths that connect the focal point system processor to target processors so that commands, messages, and alerts can flow. Two types of communication links can be selected for communication between the focal point and the target processor hardware: SNA based NetView connections (NVC) and IP based SNMP connections.</p>
Communications Task	<p>An SA z/OS user task responsible for all communications with the target processor hardware. Communications tasks receive commands from target control tasks or operator tasks and send them to the hardware. The number of communications tasks to be used is defined using the customization dialog.</p>
OSA	<p>I/O operations can display the open system adapter (OSA) channel logical definition, physical attachment, and status. You can configure an OSA channel on or off.</p>

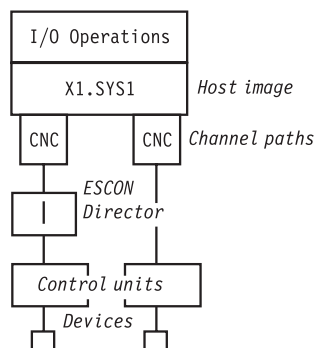
Hardware Overview

ESCON Director	The enterprise systems connection (ESCON) Directors (also known as <i>switches</i> or <i>dynamic switches</i>) are hardware electronic switching units with ports that dynamically switch to route data to input/output devices. Each port on the switch is an addressable connection, controlled by I/O operations commands. Two switches (one dynamic and one static) can be linked or chained together to increase the path length.
Ports	On an ESCON switch, a port is an addressable connection. The switch routes data through the ports to the channel or control unit. Each port can be given a name that can be entered into a switch matrix, and you can use commands to change the switch configuration.
Control units	Control units are hardware units that control input/output operations for one or more devices. You can view information about control units through I/O operations, and can start or stop data going to them by blocking and unblocking ports. For example, if a control unit needs service, you can temporarily block all I/O paths going to it.
Coupling Facilities	A hardware storage element with a high-speed cache, list processor, and locking functions that provides high performance random access to data for one system image or data that is shared among system images in a sysplex. With I/O operations you can see standalone coupling facilities. It handles them as control units with up to eight devices, all defined by the user. With SA z/OS system operations, you can display the status of coupling facilities from a single system's point of view or you can display sysplex-wide status.
Devices	Input/output devices include hardware such as printers, tape drives, direct access storage devices (DASD), displays, or communications controllers. You can access them through multiple processors. You can see information about all devices and control paths to devices. You can vary devices—or groups of devices—online or offline.
Sysplex Timer[®]	An IBM unit that synchronizes the time-of-day (TOD) clocks in a multiprocessor or in processor sides. External Time Reference (ETR) is the generic name for the IBM Sysplex Timer (9037).
NMC Workstation	A NetView feature running on a personal computer that provides a graphic series of windows controlled by NetView allowing you to monitor the SA z/OS enterprise graphically.
Automation policy	Operations and procedures (or <i>policy</i>) information is kept in files that describe routine repetitive procedures, as well as the resources in your enterprise and their relationships. Automation automatically starts, restarts, monitors, or stops resources. You define the policy using the customization dialog. The resulting files are the system operations control files (automation control file and automation manager configuration file) (see also "Automation Policy" on page 11).

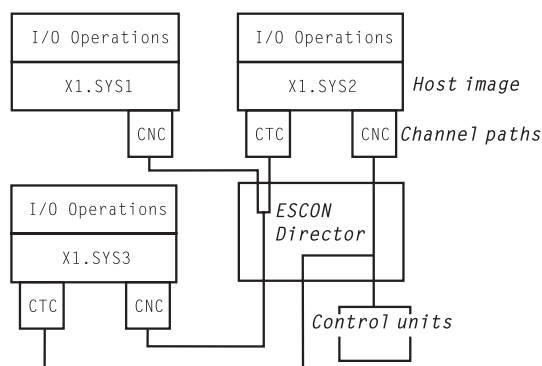
Channels

A channel is a path that transmits signals and data. It can be a standard channel or a special purpose channel such as OSA (open system adapter) or a coupling facility sender or receiver channel. For channels operating in block mode, you can display and change the state of the channels. The types of channels with ESCON switches supported by I/O operations are shown in the following list:

CNC A channel path that transfers data between a host processor image and an ESCON control unit. This path from a channel to a device may be point to point or switchable.



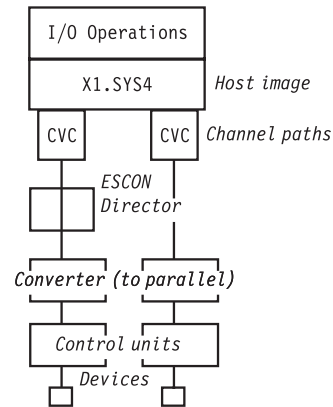
CTC The channel-to-channel (CTC) channel can communicate with a CNC on another host for intersystem communication.



CVC and CBY

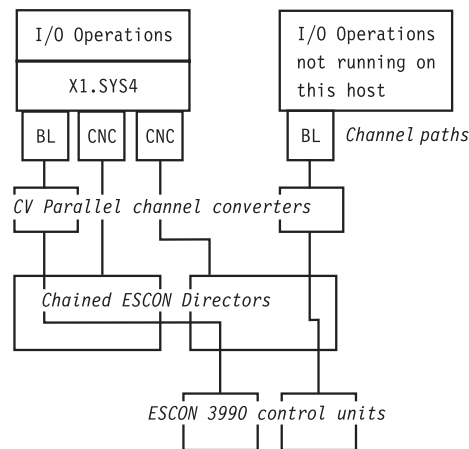
A channel operating in converted (CVC) mode transfers data in blocks and a CBY channel path transfers data in bytes. Converted CVC or CBY channel paths can communicate with a parallel control unit. This resembles a point-to-point parallel path and dedicated connection, regardless of whether it passes through a switch.

Hardware Overview



Parallel

Parallel channels operate in either byte (BY) or block (BL) mode. You can change connectivity to a parallel channel operating in block mode.



CFR/CFS and ISC/ISR

The I/O operations component of SA z/OS can display connectivity data and control connectivity for integrated system channels (ISC) connected to a coupling facility. This includes coupling facility receiver (CFR) channels and coupling facility sender (CFS) channels.

Processors, Systems

The SA z/OS processor operations component monitors the status of target processors (including logical partitions) and target systems. The status is determined based upon messages and alerts from the target hardware that processor operations has received at the focal point system.

Alerts are high-priority events that require operator action. Alerts are issued for situations in the target hardware and target system, and also for situations in the supporting processor operations environment. The following alerts will result in a status change of a resource, which will in turn cause a change in the representation of that resource in the SA z/OS displays, thus alerting you to a potential problem.

- Target hardware problem
- Wait state detected
- Disabled Console Communication Facility (DCCF) message detected
- Console lost
- Console access lost
- Communications task failed

- System console could not be obtained

For OCF-based processors, processor operations automates support element-generated alerts to update status of resources viewable on SA z/OS panels and graphic displays.

Messages received by processor operations at the focal point system allow processor operations to update the status of a target system as it initializes. You can view the initialization status of a target system using processor operations panels. Examples of some of the initialization statuses which can be seen are: Stage-1 Activate complete, IPL complete, IPL failed, Load failed, Waiting for IEA101A Start Message (z/OS target systems), and Waiting for VM Start Message (VM target systems).

I/O Devices

On a dynamically updated resource display, the status of an I/O device is updated when a message to the operator console or to the z/OS system log is trapped at the target system.

Automation Policy

Resource

Any target system, application, job, or associated hardware capable of being monitored or automated by SA z/OS.

Automation operators

Autotasks or unattended NetView tasks that do not require a terminal or logged-on operator. Automation operators may receive messages and process automation procedures.

(NetView) automation procedures

A sequence of commands, packaged as a NetView command list, a REXX procedure, or a command processor written in a high-level language. An automation procedure performs automation functions and runs under the NetView program. Automation procedures may exploit SA z/OS generic routines.

NetView automation table

A table against which the NetView program compares incoming messages. A match with an entry triggers the specified response.

Command processor

A module designed to perform a specific function. Command processors (which can be written in assembler or a high-level language) are invoked as commands.

Command list (CLIST)

A list of commands and statements written in the NetView command list language or REXX. The list performs specific functions for a user. It may be as simple as a list of commands, or complex and include variable substitution and conditional logic (more like a conventional program). Command lists are typically interpreted rather than compiled.

REXX procedure

A command list written with the Restructured Extended Executor (REXX) which is an interpretive language.

Policy database

The automation definitions (the automation policy) that the automation

Automation Policy

programmer specifies using the customization dialog is stored into the policy database. Also, using some function of the customization dialog, the automation programmer builds the system operations control files (ACF and AMC file) from the policy database. Another file built from the information contained in the policy database is the processor control file. These files derived from the policy database define how SA z/OS applies automation to the resources.

Applications

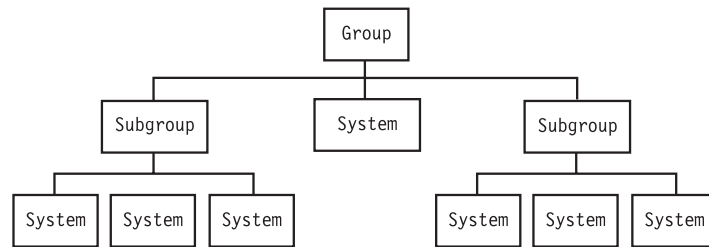
In SA z/OS, applications refer to z/OS subsystems, started tasks or jobs automated and monitored by SA z/OS. On SNMP-capable processors, "application" may be used to refer to a subsystem or process.

Application groups

Using the SA z/OS customization dialog, you can group applications into application groups. Application groups are linked to a system. Thus you can have SA z/OS automate applications that belong or depend on each other in one unit.

Grouped resource objects

Systems can be organized into groups or subgroups to simplify working with them and displaying them on panels. They can be grouped this way:



System groups can either be *standard* or *sysplex*, but processor operations treats them both the same. Applications can be grouped for being automated as an entity.

Status display facility (SDF)

An SA z/OS facility that provides a 3270-based display of automated systems and resources using assigned status colors. An operator can monitor the status of automated systems and resources by viewing the SDF main panel.

Chapter 2. Concepts

This chapter contains information that you need to know while working with SA z/OS. The following topics are discussed:

- “Automation Concepts”
- “SA z/OS Automation Flags” on page 31
- “Statuses Supplied by the Automation Manager” on page 33
- “Statuses Supplied by Automation Agents” on page 38
- “Using SA z/OS for Monitoring” on page 50

Automation Concepts

In SA z/OS, the automation function is split up:

- The observing, reacting and doing parts are located within the NetView address space, and are known as the *automation agents*. The automation agents are responsible for:
 - recovery processing
 - message processing
 - active monitoring: they propagate status changes to the automation manager
- The coordination, decision making and controlling functions are gathered into a single address space outside of NetView within each sysplex. This address space is called the *automation manager*.

You define the resources that you want to automate with SA z/OS using the customization dialog. Thus the automation manager contains a model of all of the automated resources within the sysplex. The automation agents are the automation manager’s eyes and arms. They feed the automation manager with status information and perform the actions that the automation manager tells them to. The automation manager is the brain of automation. It does all of the decision making that involves interaction between one or more resources.

The automation manager provides *sysplex-wide* automation. So its main purpose is to provide one central point of book-keeping of all resources within a sysplex, that is, the automation manager knows:

- the grouping of resources
- the dependencies between resources
- the statuses of resources
- the goals for the resources, that is when which resources are to be made available or unavailable.

According to the available information, the automation manager makes decisions and instructs the corresponding automation agent where the resource is located to put the resource into the desired state.

The automation manager knows several different statuses from each of the resources:

- The observed status
- The desired status

Automation Concepts

- The automation status
- The startability status
- The compound status
- The health status

More detail about these statuses is provided in “Statuses Supplied by the Automation Manager” on page 33.

Decision making is done by the automation manager with the help of so called goals. Goals are defined either by the automation programmer who defines permanent automation policy using the customization dialog (see *System Automation for z/OS Defining Automation Policy*) or they are created interactively by operators who issue commands to define goals.

In all cases, the automation manager is told about the goals of a certain application or resource, that is, when the resource should be up or down (available or unavailable). Goals also take into account the dependencies of resources as defined via relations in the customization dialog and they check whether a trigger is defined for a resource that determines whether the availability of a resource depends upon some external events outside of SA z/OS automation.

This is why automation performed by SA z/OS is also called *goal driven automation*.

Information about goals is provided to the automation manager in the following ways.

1 OPERATOR

By the operator command INGREQ (see “How Requests Become Goals” on page 19); these requests are labelled with originator OPERATOR(USERID) in the INGVOTE command dialog (see Figure 26 on page 95).

2 AUTOOPS

By the INGREQ command issued from automation procedure (REXX exec, CLIST); these requests are labelled with originator AUTOOPS in the INGVOTE command dialog.

3 EXTERNAL

By the OPC component; if the OPC product interfaces with SA z/OS, then OPC will take care of the startup of supporting resources required for applications that are scheduled with OPC; these requests are labelled with originator EXTERNAL in the INGVOTE command dialog.

4 SCHEDULE

There are three different ways that schedules can be used:

Resource override

That is by the operator command INGSCHED with a resource parameter, which will interactively override a service period specifically for the specified resource (see “Resource Override” on page 99). These requests are labelled with the originator SCHEDULE in the INGVOTE command dialog (see Figure 26 on page 95).

Schedule override

That is by the operator command INGSCHED, which can be used to kind of interactively override service periods for all resources that use this service period (see “Schedule Override” on page 98).

These requests are labelled with the originator SCHEDULE in the INGVOTE command dialog (see Figure 26 on page 95).

Automation policy

By means of this as defined in the customization dialog. The automation programmer uses *service periods* for applications or application groups as a comfortable method to let SA z/OS automatically schedule up and down times for these resources. Thus these service periods are also referred to as *schedules*. These requests are labelled with the originator SCHEDULE in the INGVOTE command dialog.

For most of the methods from the previous list, you can specify one of the following priorities for a goal which is checked in the case of two conflicting goals:

1. LOW
2. HIGH
3. FORCE (not applicable for schedules)

Table 3 now shows the hierarchy of all possible requests. So you can see that the strongest goal that you can specify for a resource is to interactively issue an INGREQ command to make a resource unavailable, such as:

```
INGREQ AGFG2/APL/AOC7 REQ=STOP PRI=FORCE
```

The weakest goal you can define for a resource is to make it available with low priority according to a service period.

You see that a request to make a resource available is in most cases preferred to a request to make this same resource unavailable.

Note: However, a more recent request from the same source will always overwrite an earlier request to the same resource without regard to the earlier request's priority. For example operator JOHN can overwrite his INGREQ request with HIGH priority to make a resource available with a later request of only LOW priority to make the same resource unavailable.

Note:

For all priorities except FORCE, within the same method, the request/goal to make a resource available has a higher priority than to make it unavailable.

Why then is **FORCE UnAvailable** issued by an operator of the highest priority?

This is really the only method to beat SA z/OS's attempts to make a resource available if for any reason you want to have this resource down anyway.

The operator can easily override his or her own FORCE Unavailable request for a resource with even a LOW priority request against this same resource. Thus this implementation of priorities ensures high flexibility and quick reaction in emergency cases.

Table 3. Hierarchy of Requests

Rank	Method	Goal
	FORCE	Priority

Table 3. Hierarchy of Requests (continued)

Rank	Method	Goal
1	1 OPERATOR	Unavailable
2	1 OPERATOR	Available
3	2 AUTOOPS	Unavailable
4	2 AUTOOPS	Available
5	3 EXTERNAL	Unavailable
6	3 EXTERNAL	Available
HIGH Priority		
7	1 OPERATOR	Available
8	1 OPERATOR	Unavailable
9	2 AUTOOPS	Available
10	2 AUTOOPS	Unavailable
11	3 EXTERNAL	Available
12	3 EXTERNAL	Unavailable
13	4 SCHEDULE	Available
14	4 SCHEDULE	Unavailable
LOW Priority		
15	1 OPERATOR	Available
16	1 OPERATOR	Unavailable
17	2 AUTOOPS	Available
18	2 AUTOOPS	Unavailable
19	3 EXTERNAL	Available
20	3 EXTERNAL	Unavailable
21	4 SCHEDULE	Available
22	4 SCHEDULE	Unavailable

The automation manager transforms such a goal, for example, the request that a certain resource (application) should be up, into an order to the corresponding automation agent where the application should run.

The automation agents therefore are executing the orders coming from the automation manager. While executing the automation, the automation agents also take information from the defined policy for the resources which is available in form of the automation control file on each automation agent. As an example, for an order to start a resource coming from the automation manager, the automation agents retrieve information about the appropriate startup command from the automation control file.

For enterprise monitoring, the automation manager has the task to gather and control information about which resources are available, what is the status of the resources and what status updates are occurring during automation. The SA z/OS topology manager extracts and transfers this information into the enterprise-wide RODM data base from where this information is picked up by the NMC workstation to display user-defined graphical views.

The communication between automation manager and automation agents can be implemented with WebSphere MQ. This allows you to restart of the automation manager on a different system in case the automation manager terminates unexpectedly. This ensures that no requests are lost.

The main commands you can use to retrieve information from the automation manager are:

- INGLIST: displays detail information about one or more resources (subsystem, application group and so on)
- INGINFO: displays lots of details for an individual resource or application group
- INGVOTE: displays the requests that have been issued and are currently pending for a specified resource
- INGSCHED: displays information about the current Up and DOWN service periods for resources

With the INGREQ command, you can issue a request to the automation manager, such as the goal for the specified resource should now be: available or the goal should be changed to: unavailable.

Requests that are sent to the automation manager using the INGREQ command are persistent. That means, if the automation manager terminates and is restarted later, it then remembers all requests (goals) that were valid at termination time. The automation manager will then continue to pursue all these goals for the resources. This means that if a resource should run on a certain system, and this system fails and is restarted later, the automation manager will continue to pursue the specified goals across IPL times unless they conflict with the IPL schedule times.

Summary of terminology:

Goal: There are only two goals for a resource: available or unavailable. Note that the success of carrying out a goal is also influenced by the dependencies of resources (defined as relationships in the customization dialog) and by triggers.

Note: If no goal is explicitly specified, then the default goal is: make the resource available.

Request: If an operator specifies a goal interactively using the INGREQ command, this is called a request. According to the two goals for a resource, there are the two requests to achieve these: *MakeAvailable* and *MakeUnavailable*.

Order: An order is the result of a specified goal. An operator can specify goals interactively using the INGREQ command, or an automation programmer implicitly defines goals as automation policy by means of service periods.

The automation manager transforms such a goal, for example, the request that a certain resource (application) should be up, into an order to the corresponding automation agent where the application should run.

Vote: Resources can have relationships to other resources. As an example, a dependent resource needs a supporting resource before it can be made available. See *System Automation for z/OS Defining Automation Policy* for information on how to define relationships. If an operator generates a request to start a dependent resource, the automation manager internally generates another request to make the supporting resource available as well. Or, alternatively in an INGREQ command to shutdown a resource, the operator can also request the shutdown of the dependent resources, which internally generates a request on these dependent resources. Such an internally generated request is called a *vote*.

It may happen that votes conflict or compete with each other. Such a scenario is illustrated in "Conflicting Relationships and Requests" on page 20. In such a case, the vote with the highest priority will win. If votes/requests even with the same priority come into conflict, then the requested action decides which request will win: the request to make a resource available is higher than the request to make that resource unavailable. To make a resource unavailable when it has the request to make it available, you must send it a higher priority request or revoke the *MakeAvailable* request.

Using the INGVOTE command, you will see all currently active requests with their priorities, may they have been generated explicitly by an operator, implicitly because of the automation policy or internally as a consequence of relationships.

One automation manager must exist on each sysplex and one on each standalone system. In a sysplex, you can have one or more secondary automation managers (SAM) for backup purposes. That is, if the system on which the primary automation manager (PAM) runs, abends or if the automation manager itself abends, the first secondary automation manager recognizing this, will become the

new PAM and will take over all current requests from the former PAM. The automation agents in the sysplex will not notice the switch from one automation manager to the other.

Note:
No request will be lost.

How Requests Become Goals

A request asks that one specific resource (its *target resource*) should be moved to a specific state (its *goal*). A resource may, for example, be an application that should run on a certain system.

Example:

```
INGREQ abc REQ=START
```

specifies that SA z/OS should start the resource called *abc*, that is, SA z/OS should set the desired status of this resource to AVAILABLE on a system.

Now, SA z/OS knows the dependencies for each of the actions that can be taken against the resource through the resource's relationships with its supporting resources. Using these, it propagates the request onto those resources. These internally created requests are called *votes*. Let us have a look at a CICS scenario: It makes sense to start a *Terminal Owning Region* (TOR_1) only after an *Application Owning Region* (AOR_1) is up. So you could define TOR_1 (running on a certain system) in our example to become a dependent resource and AOR_1 to become the supporting resource. So if you request to start TOR_1, SA z/OS would generate a vote to start AOR_1.

Start requests are generally propagated from dependent resources up to supporting resources so that the supporting resource is started before the dependent resource is started. Stop requests are propagated down to dependent resources in order to stop dependent resources first.

How Request Propagation Works

If the request is a higher priority than whatever request the supporting resource currently has, then the new request wins and the supporting resource's goal is changed to comply with the request. The process then repeats itself for the supporting resource to its supporting resources.

Eventually the propagation will stop either if SA z/OS does not encounter another supporting resource or when it encounters higher priority requests. At this point, SA z/OS knows which resources are not in their goal state. For those whose prerequisites for being started or stopped are met, SA z/OS will issue start or stop commands.

Once a resource's goal has been set, SA z/OS will continue trying to achieve that goal. This means that if the resource abends, and its desired status is still AVAILABLE, then SA z/OS will try to restart it. If the goal changes to UNAVAILABLE, SA z/OS will stop the resource. If a resource starts when its goal is UNAVAILABLE, SA z/OS will also stop it.

Conflicting Relationships and Requests

One of the most effective automation capabilities of SA z/OS is the concept of *dependency relationships*. Using the customization dialog, the automation programmer defines relationships and conditions between dependent resources and supporting resources. For information on how to achieve this, refer to *System Automation for z/OS Defining Automation Policy*.

In complex environments, it may happen that relationships exist that are opposed in both direction and meaning as illustrated in Figure 2. The following abbreviations are used:

- for dependency relationships

MaWa denotes *MAKEAVAILABLE/WhenAvailable*

MuWu

denotes *MAKEUNAVAILABLE/WhenUnavailable*

- for statuses

Unav denotes *Unavailable*

Avail denotes *Available*

- for requests or votes

Av denotes a request or vote to become available

Un denotes a request or vote to become unavailable

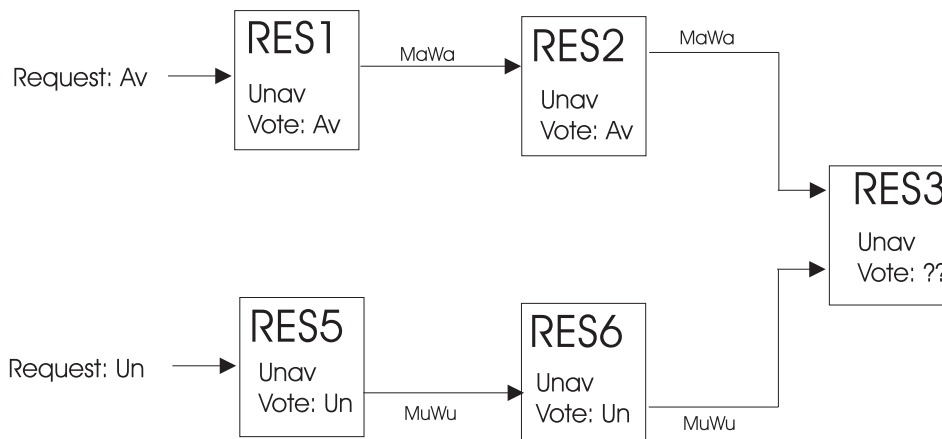


Figure 2. *Conflicting Relationships and Requests*

In Figure 2, *RES1* is in status Unavailable (Unav) and gets the request to become available (which internally creates the corresponding vote).

So the relationship *MAKEAVAILABLE/WhenAvailable* propagates requests to make resources available and the relationship *MAKEUNAVAILABLE/WhenUnavailable* propagates requests to make resources unavailable.

Note: The first part of a dependency relationship always applies to the resource where the arrow starts, and the condition always applies to the resource where the arrow ends.

As requests are internally stored and propagated as votes, in Figure 2, you see two conflicting votes that would be propagated to *RES3*. From *RES1* we have the vote

to be available while from *RES5* we have the vote to be unavailable. *RES3* has received both of these votes, but cannot satisfy them at the same time. You could see the conflicting votes for *RES3* by issuing the *INGVOTE* command against this resource.

In cases when two separate streams of propagation meet, the priority of votes/requests is evaluated and used to determine the winning vote. A priority is assigned to a request either by the operator interactively using the *INGREQ* command or by the automation programmer setting priorities for service periods.

A request to make a resource available wins over a request to make it unavailable with the same priority. This is only valid due to the *propagation* of requests. However, if an operator sends a request to start a resource, and the same or another operator later sends another request to stop this resource, the latter one will win. The difference is, that this is not a *propagation* of requests.

How the Automation Manager and Automation Agents Cooperate in a Sysplex

As described in “Automation Concepts” on page 13, SA z/OS’s automation logic in a sysplex is split up between the automation agents and the automation manager.

Also, for reasons of high availability, SA z/OS offers the concept of the primary and secondary automation manager (PAM and SAM), where the SAM can take over the work of the PAM without loss of any information or without any requests being processed twice.

In a sysplex, the automation agents communicate with each other exploiting XCF. Automation managers communicate with the automation agents exploiting WebSphere MQ or XCF.

SA z/OS provides two options for establishing communication between the automation manager and the automation agents:

- Using WebSphere MQ V5R2 queues for both communication and backup
- Using XCF for communication and a VSAM data set for backup

Figure 3 on page 22 shows a typical configuration in a sysplex, showing the communication paths and methods between its members. Automation managers and automation agents communicate via WebSphere MQ using the coupling facility. Automation agents communicate via XCF. XCF is only used at initialization time or to determine if an automation agent leaves or joins the sysplex. For more detailed information on how WebSphere MQ queues are used to perform the communication refer to *System Automation for z/OS Planning and Installation*.

Controlling Application Move and Server Management

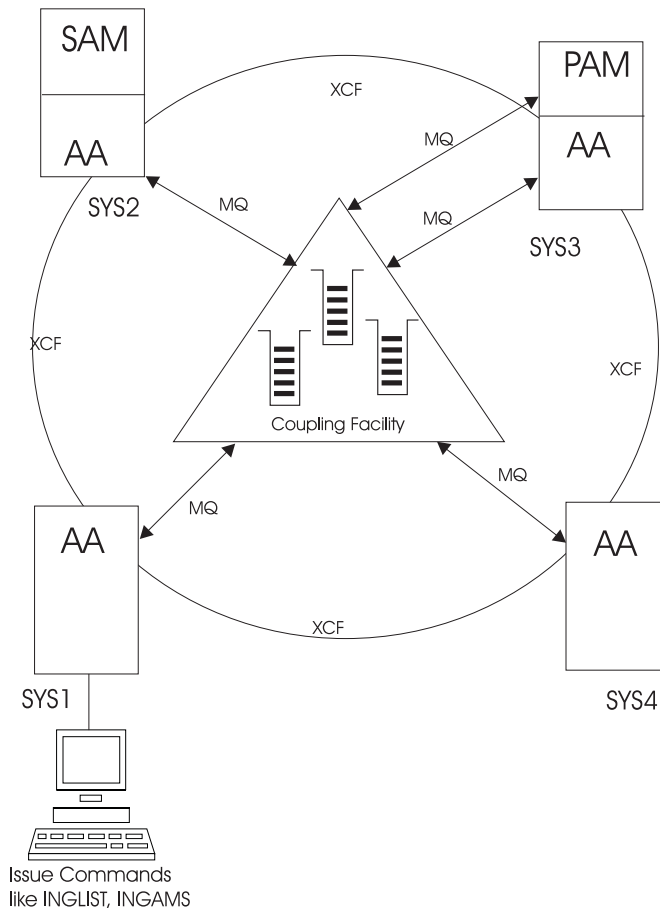


Figure 3. Automation Manager and Automation Agents Cooperation in a Sysplex

Controlling Application Move and Server Management

Application move refers to the process of stopping an active set of resources and starting an alternative set in a coordinated fashion.

Server management refers to the starting and stopping of equivalent resources to maintain a desired level of server capacity. Although it will work with any type of resource, it is designed to work with servers that are loosely coupled to the applications using them. Loosely coupled in this sense means that applications using the servers will not be adversely impacted if one instance of the server group is stopped and another one is started.

SA z/OS uses the mechanism of grouping resources as described in *System Automation for z/OS Defining Automation Policy* to implement the move mechanism, namely the groups of nature MOVE and SERVER obey all move instructions described later in this section, along with an availability target instruction that specifies, how many members of a group should be available, if not all are required.

Move Functions

The move functions that you can specify against a group are:

- Automatic group management to maintain application availability. This includes the activation of backup/alternate resources in response to resource or system failure.

Controlling Application Move and Server Management

- The ability to move resources away from a system en masse, on either a group/subgroup level or a system level.
- The ability to query and dynamically change a group's policy so as to cause one instance of a resource to be stopped and another started.
- The ability to estimate the impact of those changes before committing them.
- The ability to perform actions resulting from the above in a delayed, non-disruptive fashion.
- The ability to return resources to their normal systems in a delayed, non-disruptive fashion.

The INGGROUP command offers the following methods to apply move functions to a group:

EXCLUDE/AVOID/INCLUDE

This indicates that you want to remove movable members from a system (exclude), prohibit SA z/OS from activating movable members on a system (avoid) or that you wish to undo either of the other two actions (include).

Availability Target (AVTGT)

This specifies the availability target that is to be set for the group. The value is the number of the group members that should be made available when the group has to be available. You can dynamically overwrite a group's availability target which is statically defined in the automation policy.

Preference (PREF)

This parameter specifies preference values for being available for the members of a group. You can dynamically overwrite the preference values which are statically defined in the automation policy for each member. More information about the preference value is provided in "Controlling Application Move with Preference Values."

Satisfactory Target (SATTGT)

This parameter specifies the number of members in the group that must be active before the automation manager treats the group as being in a satisfactory state. The value must be a positive, decimal number. The parameter will be ignored for groups other than server groups. You can dynamically overwrite a group's satisfactory target which is statically defined in the automation policy.

Controlling Application Move with Preference Values

Controlling application move and server management is done by means of preference values. Preference values are numbers between 0 and 3200. They express the importance of each individual member of the group.

Note that only MOVE groups and SERVER groups support preference values. With BASIC groups you cannot specify preferences, as all members need to be available. For MOVE and SERVER groups, the preference defines which of the resource members of the group should preferably run. Remember that a MOVE group is designed for backup solutions. A MOVE group can have many components, but one and only one is supposed to be active. All other components are for backup only. Thus MOVE groups usually contain members of one kind (where one application can replace the others).

Controlling Application Move and Server Management

SERVER groups are created when not all but only a certain number of their resources need to be available in order to consider the application group as being available. The number of desired available resources is controlled by the *Availability Target* parameter in the customization dialog. And once again, with the preference value in the customization dialog, you control which resources you prefer to be available and which you want to keep as backups.

Note that preference values are statically defined with the automation policy defined in the customization dialog for the groups. Using the INGGROUP command, operators can interactively change the preferences of group members. This new preference adjustment replaces the value defined in the customization dialog.

A special threshold preference value is 600. This acts as a divider between resources that are available for normal startup (primary instances) and resources with lower preferences that are only available for startup during recovery situations (backup instances).

Four preference values are maintained by SA z/OS for each member in a group:

1. The *base preference* which is specified in the customization dialog.
2. A *preference adjustment* which is dynamically set via the INGGROUP command.
3. Its *calculated preference* which is the sum of its base preference and its preference adjustment.
4. Its *effective preference* which is the sum of its calculated preference and some modifications that the automation manager applies.

The value of the *effective preference* will be visible in the INGGROUP command dialog as shown in Figure 40 on page 111.

The automation manager applies the following modifications to the preference values:

- If the resources observed status is Unknown, SysGone, HardDown or Problem its effective preference is always 0 and the manager will never attempt to start it.
- If the resource requires an excluded system, its effective preference is always 0, because it will not be able to receive requests anyway.
- If the resource requires an avoided system and it is not available, then its effective preference has a maximum value of 100 and it can only be picked in recovery mode (see “Recovery Mode” on page 25).
- If the resource is Available or WasAvailable, Starting or Degraded, its effective preference is increased by 225.
- If the resource is selected, its effective preference is increased by a further 25.
- The effective preference is decreased by 400 when the observed status SysGone and the base preference is greater than 1500.

This lets the automation manager calculate an effective preference for each member that reflects both the users preference and the current state of the member.

Note: If a base preference 0 has been specified for a resource of a SERVER or MOVE group in the customization dialog, and it is in one of the observed statuses: *Available*, *WasAvailable*, *Degraded*, *Starting* then SA z/OS grants an effective preference of 1250 upon this resource.

Recovery Mode

In this context, the concept of the recovery mode is of importance. Recovery mode is entered when a previously available member of the group goes into an unavailable status and the group is unable to find enough members with a *calculated preference* greater than 600 to make the group available. When this happens it enters recovery mode and is allowed to start members with a calculated preference of less than 600.

Initially, when the group is not in recovery mode, only members with a calculated preference of greater than 600 will be started. Once the group is available, it will try to maintain its availability by using only members with a calculated preferences of greater than 600. If it cannot, then it will enter recovery mode and use the lower preference members to bring the group back to availability.

The preference value and recovery mode interact to allow you to specify the behavior of members of the group.

Note: Recovery mode is automatically entered if an EXCLUDE list is used with the INGGROUP command (see also “Move Functions” on page 22).

Available Preference Values

See Table 4 for available preference values. Preferences other than those listed are also accepted to define your own fine-tuning.

Note: The preference values listed in Table 4 are *calculated preferences*.

Table 4. Preference Values and Behavior of Members

Calculated Preference	Resulting Behavior
Values for Primary Resources	
>2600	The resource is always selected regardless of its state. Use this value if the resource must run.
>2000	The resource is always selected unless the observed status of the resource is SysGone or HardDown. If it is one of these statuses then it is only selected if no other viable member exists. A viable member is a resource that can be started and that is allowed to be started. In principle, the observed status is one of the following: SoftDown, Starting, Degraded, Available or WasAvailable.
1000	The member is always selected when the group has been started and the system associated to this member comes up. Other resources from the same group with preference 700 are stopped in favour of this member.
900	The member is always selected when the group has been started and the system associated to this member comes up. Members with preference 500 will be stopped in favour of this group.
700	The member is always selected when the group has been started and the system associated to this member comes up. Members with preference 300 will be stopped in favour of this group. It is not stopped unless a very high preference (1000) alternative becomes available. This value is the default. Use it in a MOVE group for the resource that normally runs.
<600	This value is used as a threshold. A resource with preference 600 and lower will not be started at group startup. Only picked if the group is in recovery mode.

|
|
|

Controlling Application Move and Server Management

Table 4. Preference Values and Behavior of Members (continued)

Calculated Preference	Resulting Behavior
Values for Backup Resources	
500	This resource is not selected but will be selected when the group enters recovery mode. Once the problem is over, it will be stopped as soon as a primary member with a calculated preference > 750 is available.
300	This resource is not selected but will be selected when the group enters recovery mode. Once the problem is over, it will be stopped as soon as a primary member with a calculated preference > 550 is available.
100	This resource is not selected but will be selected when the group enters recovery mode. As soon as a preference 500 backup member is available, it will be stopped in favour of the backup member.
1	A resource with this preference will never be selected by SA z/OS automation; it is always sent an Offline vote. Operators may make such resources available interactively by using the INGGROUP command and give it a higher preference. Use this value to provide completely manually controlled application availability.
0	This value indicates that the member is passive and will never have a vote propagated to it. If a member is made passive, then any votes that have been propagated to it will be withdrawn.

Preference Bonuses

The preference bonuses given in Table 5 are used to enable a greater degree of control of resources that have a high preference value (above 1500).

Table 5. Preference Bonuses

Bonus	Reason
+175	Member has variable selected/group_name set to 1 and has base preference > 1000.
+25	Member has variable selected/group_name set to 1 and has Base preference <= 1000.
+225	Member is Starting, Degraded, Available, Stopping
-400	Member is SysGone and has base preference >1500

Preference Scenarios

To understand the interactions of the various flavours of preference values and the concept of the recovery mode, the following sections describe some scenarios of how SA z/OS reacts according to the specified and automatically applied preferences.

Backup with Non-Disruptive Restoration: In this scenario you have an application *ABC* which can run on one of two systems - *SYS1* and *SYS2*. Only one instance of the application should normally be active. The instance on *SYS1* is the primary instance (set preference 700), while the instance on *SYS2* is the backup instance (set preference 500).

The preferences 700 and 500 mean that the applications will behave as follows:

1. If *SYS2* is IPLd first, then the application will not be started (preference 500, not in recovery mode).

When *SYS1* is IPLd, its instance of the application will be started (preference 700, highest startable instance).

Controlling Application Move and Server Management

2. If *SYS1* should fail, the instance on *SYS2* will be started (preference 500, in recovery mode, highest startable instance).
3. When *SYS1* is IPLd again, it will not start its instance (preference 700, but active instance with preference 750 (500 + 250 for being available)).
4. At the end of the service period the instance of *SYS2* will be stopped.
5. At the start of the next service period, the instance of *SYS1* will be started (preference 700, highest startable instance).

Backup with Disruptive Restoration: This scenario is like the above, but you want the application to return to *SYS1* as soon as *SYS1* is available, even if this causes an additional service interruption. You can use preferences of 700 for *SYS1* and 300 for *SYS2*, or 900 for *SYS1* and 500 for *SYS2*. As these are the only two applications involved, either combination will work (the important thing is that they are more than 250 apart). The example uses the 700 and 300 values.

1. If *SYS2* is IPLd first, then the application will not be started (preference 300, not in recovery mode).
When *SYS1* is IPLd, its instance of the application will be started (preference 700, highest startable instance).
2. If *SYS1* should fail, the instance on *SYS2* will be started (preference 300, in recovery mode, highest startable instance).
3. When *SYS1* is IPLd again, its instance will be started (preference 700, highest startable instance) and the *SYS2* instance will be stopped (preference 550 (300 + 250 for being available), lower than 700). This causes a short service interruption as the application moves back to *SYS1*.
4. At the end of the service period the instance of *SYS1* will be stopped.
5. At the start of the next service period, the instance of *SYS1* will be started (preference 700, highest startable instance).

Figure 4 on page 28 illustrates the difference of non-disruptive and disruptive restoration between the steps 2 and 3 in each scenario: in the non-disruptive scenario, the application *ABC* continues to run on *SYS2* until the end of the service period for the group, because this resource now has a higher priority of 750 than the resource on *SYS1* which would only have priority 700. In the disruptive restoration on the other hand, *SYS1* takes over for the rest of the service period after the problem has been fixed.

Controlling Application Move and Server Management

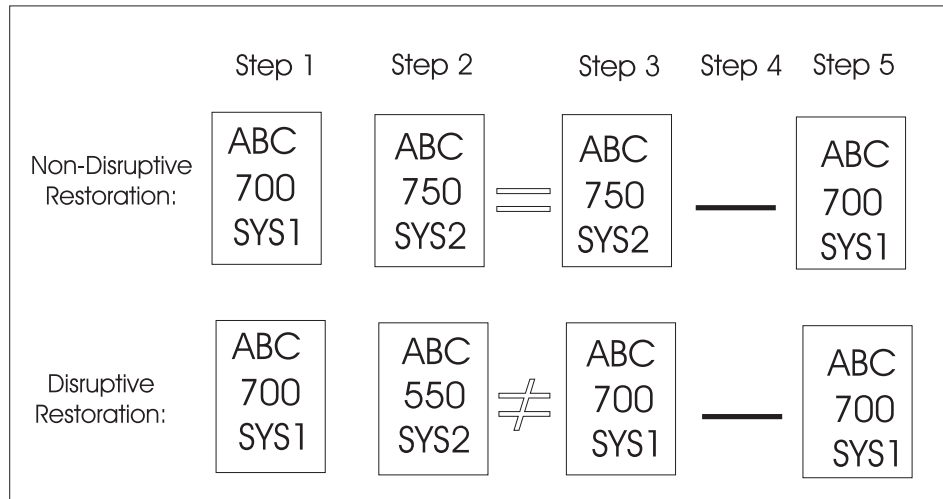


Figure 4. Non-Disruptive and Disruptive Restoration

Real Hierarchy of Preferences: In this scenario, you have three systems where the application can be run. *SYS1* is where you normally run it, *SYS2* is a back up and *SYS3* is where it can run in extreme emergency only. You would specify preferences as 700, 500 and 100.

1. If either *SYS2* or *SYS3* are IPLd first, their instances will not be started (preferences 500 and 100, not in recovery mode). The operator can, of course, explicitly request one of the instances to be started (which should be done if they want the application to be available and know that *SYS1* will not be IPLd for a while).
2. When *SYS1* comes up its instance will be started (preference 700, highest startable instance).
3. If *SYS1* fails, the *SYS2* instance will be started (preference 500, in recovery mode, highest startable instance).
4. Now, if *SYS1* were to return at this point, we would be back in the first scenario - non-disruptive backup.
5. Instead, let us suppose that *SYS2* now fails. This causes the instance on *SYS3* to be started (preference 100, in recovery mode, highest startable instance).
6. Now if either of the other two systems came back (let us assume it is *SYS2*), then its instance will be started (preference 500, in recovery mode, highest preference instance) and the instance on *SYS3* will be stopped (preference 350 (100 plus 250 for being available), lower than 500). This would cause an application outage, but would get the application off *SYS3*.
7. If *SYS1* then comes back, its instance will not be started—preference 700, lower than 750 (500 + 250 for being available).
8. After the next service period stop, we would be back to running the instance on *SYS1*.

Note: If *SYS1* was down at the end of the start of the next service period, then the application would not be restarted. Using preferences of 900, 700, 100 (or 300) would get around this, but also means that the application could be automatically started on *SYS2* if it was IPLd before *SYS1*.

Server Battery with Backups: In this scenario your sysplex consists of four systems (*SYS1*, *SYS2*, *SYS3* and *SYS4*). You defined a Sysplex Application Group of nature SERVER with an availability target of 3 that should normally run on

Controlling Application Move and Server Management

systems *SYS1*, *SYS2* and *SYS3*. A backup resource is defined on *SYS4*, but it should only be used when one or more of the main servers is unavailable. Preference of 700 should be used for the resources on *SYS1*, *SYS2* and *SYS3*, while the backup resource on *SYS4* should have a preference of 300.

1. When all systems are IPLd, the resources on *SYS1*, *SYS2* and *SYS3* will be started (preference 700, 3 highest startable resources). The resource on *SYS4* will not be started (preference 300, not in recovery mode).
2. If either *SYS1*, *SYS2* or *SYS3* should fail, then the resource on *SYS4* will be started (preference 300, now in recovery mode).
3. When the failed system is restored, its resources will be restarted—preference 700, three highest startable resources—and the resource on *SYS4* will be stopped—preference 550 (300 + 250 for being available).

Note: Setting a preference of 500 for the resource on *SYS4* would cause it remain running when the failed system returned, effectively turning the resource on the returned system into the back up. It is also possible to set the preferences as 700, 700, 700, 690 which will let automation start the resources on which ever systems it needs to make the application available, but to prefer not to use the *SYS4* resource if the other three are available (which they may not be at IPL time).

Move Groups and Recovery Mode: In this scenario your sysplex consists of four systems (*SYS1*, *SYS2*, *SYS3* and *SYS4*). You defined a Sysplex Application Group of nature MOVE (availability target of 1) that should normally run on system *SYS1* (preferences 700, 450, 250, and 1 respectively).

1. When all systems are IPLd, the resource on *SYS1* will be started (preference 700, highest startable resource). The resources on *SYS2*, *SYS3*, and *SYS4* will not be started (availability target of 1 and APG not in recovery mode with preferences ≤ 600).
2. If *SYS1* should fail then we enter recovery mode, which means that *SYS2* and *SYS3* become eligible to be started. In this case *SYS2* will be chosen (highest eligible preference 450). If however, *SYS1* was a planned shutdown (SHUTSYS ALL) then we would not be in recovery mode and neither *SYS2* nor *SYS3* would be eligible, so the application would not be moved automatically. To move the resource to another system use INGGROUP to raise that system's preference to > 600 .
3. If *SYS2* should fail, as well as *SYS1*, then the resource will be started on *SYS3* without further adjustment (minimum startable preference 250).

Note: *SYS4* has not participated in any of these MOVE examples (preference value 1, non-startable instance).

Absolute manual control: Member to be activated gets preference 3200, all others get preference 1.

The member with the 3200 preference will be selected. All others will always have a MakeUnavailable vote propagated to them.

This does not change even if the system with the selected member leaves the sysplex.

Move Only After System Failure: Primary member defined preference 2800, backups preference 500.

Controlling Application Move and Server Management

As long as the primary is not SYSGONE it will be selected. If the primary is SYSGONE, its preference drops to 2575 ($2800 - 400 + 175$), allowing an alternative viable member to be selected.

The backups will only be viable if the resource was previously active because of the groups recovery mode.

Move Only If System Down: Primary member defined preference 2800, backups preference 700.

As long as the primary member is not SYSGONE it will be selected. If the primary member is SYSGONE, its preference drops to 2575 ($2800 - 400 + 175$), allowing an alternative viable resource to be selected.

The backups are always viable.

Sticky Move Only After System Down: Primary member defined preference 2801, backups preference 2800.

As long as the primary member is not SYSGONE, it will be selected. If the primary becomes SYSGONE, its preference drops to 2576 ($2801 - 400 + 175$), allowing one of the backups to be selected. The sticky bonus raises its preference to 2975 ($2800 + 175$), meaning that even when the primary returns the selected backup system will remain selected.

It will only be deselected if it goes to SYSGONE, in which case the primary will be re-selected if it is available otherwise another backup would be picked.

Move Only After System or Application Failure: Primary member defined with preference 2400, backups preference 500.

As long as the primary member is not SYSGONE or HARDDOWN it remains selected. If it is HARDDOWN or SYSGONE its preference is still <2600 , so a viable alternative can be selected.

The backups are only viable if you are in recovery mode (that is, the member was active when the failure occurred).

Move Only If System Down or Application Failure: Primary member defined with preference 2400, backups preference 700.

As long as the primary member is not SYSGONE or HARDDOWN it remains selected. If it is HARDDOWN or SYSGONE its preference is still <2600 , so a viable alternative can be selected.

Sticky Move If System Down, Move After Application Failure: Primary member defined with preference 2401, backups preference 2400.

As long as the primary is not SYSGONE or HARDDOWN it remains selected. If it is SYSGONE or HARDDOWN its preference is <2600 , so a viable alternative may be chosen.

The sticky bonus raises the preference of the selected member to 2575 ($2400 + 175$), so it will remain selected until gets to a state of SYSGONE or HARDDOWN, at which point another viable alternative will be selected, with the primary member having preference.

The individual shutdown of the active member will not trigger a move.

Sticky Move Group, Unless Member Stopped: Primary member defined with preference 701, backups preference 700.

As long as the primary member is startable, it remains selected. If it becomes non-startable, another member is selected and that remains selected as long as it is startable.

Individually stopping the active member will trigger a move. If the members were defined with preferences >1500, the bonus would be 175 points rather than 25 points.

Sticky Move Group, Until Told Otherwise: Primary member defined with preference 3200, backups with preference 1600.

With base preferences of 3200, you get it picking a member when the sysplex is IPLd and retaining that one until an operator uses INGGROUP to make it pick another one.

Beware of timing problems as the first system with a member to register will probably get it.

SA z/OS Automation Flags

Automation flags are automation policy settings defined in the customization dialog that specify whether certain types of automation are on or off for a resource. For information on how to set automation flags, refer to *System Automation for z/OS Defining Automation Policy*.

Automation Agent Flags

If an event occurs that triggers automation, SA z/OS checks the appropriate flag to determine if automation is currently on. If it is, SA z/OS proceeds with the automation actions associated with the flag. For example, if TSO fails and the Restart flag is on for TSO, SA z/OS restarts TSO according to the policy specified in the customization dialog.

There is one flag that controls all the other flags. This is:

Automation

If this flag is set to N (No), no automation is allowed, regardless of how the other flags are set. If this is set to Y (Yes), SA z/OS checks each of the other flags to see if they are set to Y or N.

The other flags which apply to each resource are:

Initstart

If this flag is turned on and the SA z/OS automation status for a subsystem is DOWN, SA z/OS can try and start the subsystem when all other conditions necessary for its startup (as defined in your automation policy) are met.

Start If this is on, automation manages and completes the startup process of the subsystem.

Recovery

If this is on, automation takes the specified action for any MVS message that is not associated with the initialization, startup, shutdown, or restart

SA z/OS Automation Flags

of a subsystem. For example, if this flag is on, automation takes corrective actions in response to messages that indicate shortages of WTO buffers and JES spool space. Automation also invokes offload routines when the following MVS data sets are about to fill up: SYSLOG, LOGREC, SMF, MVS dump.

Shutdown

If this flag is turned on, SA z/OS can process automated shutdown requests for this subsystem. The setting of this flag can be overridden when the shutdown is requested. Also, the flag can be used to control secondary automation that occurs during the shutdown process. Secondary automation covers things such as replying asynchronously to WTORs and issuing commands in response to messages from the subsystem.

Restart

If this flag is turned on and the SA z/OS automation status for a subsystem is RESTART, SA z/OS can try and start the subsystem when all other conditions necessary for its startup (as defined in your automation policy) are met.

When you set an automation flag to Y in the customization dialog, that automation flag is on all the time, unless someone does one of the following:

- Turns the flag off for a period by scheduling an override in the customization dialog. For example, you could schedule an override that turns initialization and shutdown automation off on Sundays from 12:00 to 16:00, so you can do your system backups.

You can display these scheduled overrides using the DISPSCHD command dialog, as explained in “Displaying Automation Agent Flag Override Settings” on page 105.

- Turns the automation off using the INGAUTO command dialog. These command dialogs are discussed later in the chapter.
- Causes a higher level flag to be turned off, for example the subsystem’s Automation flag or the SUBSYSTEM Restart flag.

If a flag is set to N, you cannot schedule an override in the customization dialog to set it on for a particular time. You must use the INGAUTO command dialog.

If a flag is set to E, then an exit is executed to determine whether or not automation is allowed. See *System Automation for z/OS Customizing and Programming* for details of flag exits.

The Automation Manager Global Automation Flag

Using the INGLIST or the INGSET command (see *System Automation for z/OS Operator’s Commands*) you can set an automation flag for the individual resources, which is checked by the automation manager before it sends any order to the automation agent to start or stop the specific resource.

The purpose of this flag is to prevent (if flag is set to NO) or enable (YES) the starting or stopping of resources from the point of view of the automation manager. This can be done for resources that reside on systems that are currently inactive, for example, to prevent the startup of the resource at IPL time of the system.

The Hold Flag

As the default for all resources is to be available, you can use the *hold flag* for resources to prevent their startup at IPL time of the system.

The hold flag is primarily used for the NOSTART reply to the AOF603D WTOR.

You can manipulate the hold flag with the INGLIST or the INGSET command or just view it with INGINFO.

Statuses Supplied by the Automation Manager

The automation manager keeps track of various statuses for each resource:

Observed Status

This indicates the current status of the resource, as reported by the automation agent. See “Observed Status” on page 34.

Automation Status

The automation status of a resource tells the automation manager in which automation status a resource currently resides. If a resource’s automation status currently is *idle*, this means that the automation agent does nothing for automation, or conversely, if the automation status is *denied*, this means that automation is currently not allowed, possibly because the automation flag has been set off. See “Automation Status” on page 35.

Desired Status

This status reflects the goal of a resource. The automation manager will be continuously trying to place the resource into this status unless the goal is revoked or changed. This status can be either *Available* or *Unavailable*.

If it is *Available*, the automation manager will try and start the resource. If it is *Unavailable*, the automation manager will try and stop the resource. The default is to try and make the resource available. See “Desired Status” on page 36.

Startability Status

This is a calculation as to whether or not it is possible to start the resource if this is requested. It includes the startability of all inactive start dependent resources. This status is mainly used as input for Compound Status calculations and by groups to prevent them from selecting members that cannot be started. See “Startability Status” on page 36.

Compound Status

This status is the composition of the other statuses to give the operator a chance to check from the automation manager if a resource is currently OK concerning the automation. See “Compound Status” on page 37.

Health Status

With SA z/OS, application-specific performance and health monitoring has been introduced. The health status can be used by the automation manager to make decisions and, if necessary, trigger automation for the application if there is a change in the health state. See “Health Status” on page 37.

The operator can retrieve the status using two commands:

- INGLIST command: provides some main attributes of a resource, for example a resource’s different statuses.
- INGINFO command: provides information about a resource from the point of view of the automation manager.

Statuses Supplied by the Automation Manager

Three of these statuses, the observed status, automation status, and Health status are supplied by the automation agent responsible for running the resource. The desired status is derived through the request propagation mechanism. The startability status is derived through status propagation. The final status, compound, is a summary of all of the other statuses and a number of other values. The compound status is made visible through INGLIST or INGINFO and, eventually, placed into RODM and made visible on the NMC workstation.

Observed Status

This represents the current status of the resource and is provided by the related automation agent.

Possible values are:

Unknown (Assumed unavailable)

The automation manager has no observed status information about the resource. To be on the safe side, the automation manager will assume that the resource is unavailable and will try to make it available if this is the desired status. On the other hand, this assumption allows the automation manager to process shutdown requests for supporting resources which can only be made unavailable if this resource is unavailable.

SysGone (unavailable)

The system on which the resource is defined is no longer a member of the sysplex.

SoftDown (unavailable)

The resource is unavailable and automation may restart it.

HardDown (unavailable)

The resource is unavailable and automation will not restart it.

Starting (available)

The automation agent is either in the process of making the resource available or has detected actions from some outside command to make the resource available.

Available

The resource is ready for use.

Degraded (available)

The resource is available but not all required members in the server group are available.

Standby (available)

The resource has a primary/secondary system association defined. The automation agent posts this status to the automation manager when setting the automation agent status to MOVED or FALLBACK.

Note: The automation manager treats STANDBY like a HARDDOWN status except that it is not considered that this indicates an error condition.

Problem (available)

The resource is available but has a serious problem, probably making it unusable.

Stopping (available)

Either the automation agent is in the process of stopping the resource, or it has detected that some outside command is doing so.

Statuses Supplied by the Automation Manager

WasAvailable (Assumed available)

The automation manager has lost contact with the automation agent responsible for the resource, however the resource was available when contact was lost and the system it is on still appears to be running. The resource will be treated as being available. This is primarily to prevent inappropriate recovery actions from being initiated whenever an SA z/OS NetView is recycled.

Observed Status Transitions

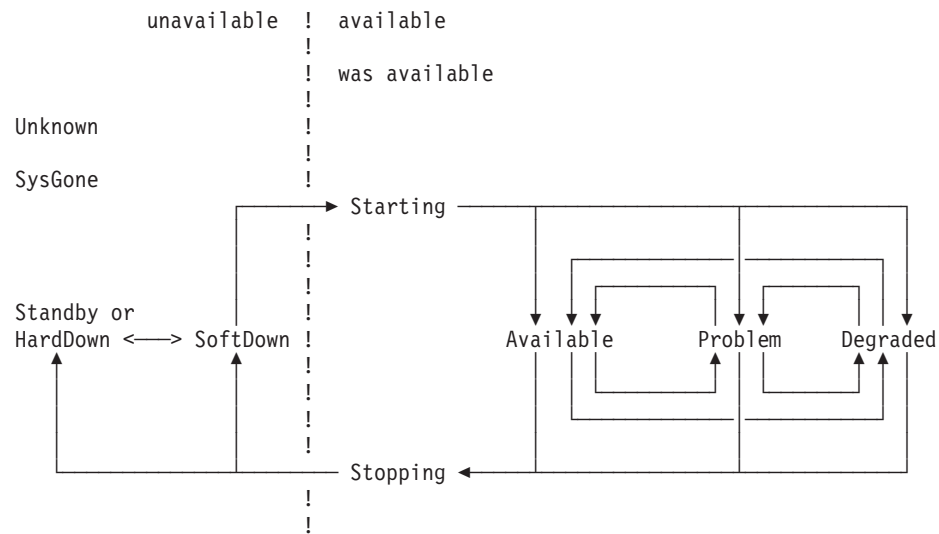


Figure 5. Observed Status Transitions

Automation Status

This indicates the status of the automation for the resource. It is provided by the related automation agent.

Possible values are:

Unknown

No connection with the automation agent.

Idle

No orders have been sent; the automation agent is not doing anything. The manager may send the automation agent new orders.

Ordered

Orders have been sent. This status is only posted within the automation manager.

Busy

The automation agent is processing the orders, or observing/assisting a start or stop process initiated by some outside command. This status is entered when a resources status becomes Starting or Stopping and the previous Automation Status was Idle or Ordered. This status is changed to Idle if the resource achieves an observed status other than Starting or Stopping. This status may be posted directly by the automation agent.

Denied

The automation agent was not allowed to process the last order it received. This status is changed to Idle if the resource achieves an observed status of HardDown, SoftDown or Available. This status is posted directly by the automation agent.

States Supplied by the Automation Manager

Problem

The automation agent encountered a problem while processing for the resource. This status is changed to Idle if the resource achieves an observed status of HardDown, SoftDown or Available. This status is posted directly by the automation agent.

Internal

This means that the automation of the resource is being handled internally.

Automation Status Transition

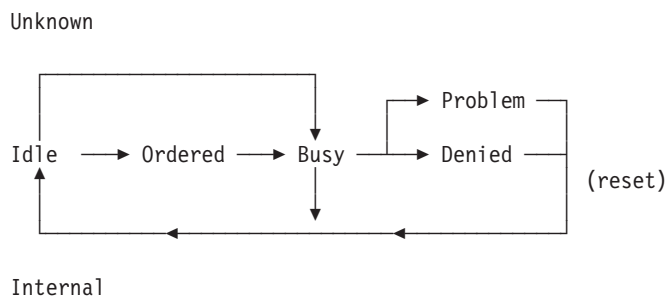


Figure 6. Automations Status Transitions

The automation manager may only send orders to an idle automation agent.

Transition from Idle to Busy occurs when the automation agent detects a third party (such as an operator or ARM) starting or stopping a resource.

Desired Status

You can set the desired status of a resource either:

- With the INGREQ command
- With a service period
- With a schedule or resource override (INGSCHED)
- Or indirectly through a vote which stems from the request propagation

Possible values are:

Available

The resource should be started (made available).

Unavailable

The resource should be stopped (made unavailable).

Startability Status

The startability status is calculated from the resource's current status, its automation status, its automation and hold flags and the startability of its supporting resources.

Possible values are:

Yes The resource is either startable or already started.

No The resource cannot be started due to a problem with the resource itself.

Inhibited

The resource cannot be started because of either a problem with one of its supporting resources or because automation has been prohibited.

Denied

The resource cannot be started because the automation status of this resource is *Denied*.

Compound Status

This status is a summary of all the statuses of a resource. It considers elements of the four other statuses as well as a number of other values.

Possible values are:

Problem

There is a problem with this resource that automation cannot solve. Operator intervention is required.

Denied

The resource is not in its desired state and automation is unable to proceed because either the automation flag has been turned off or the automation manager hold flag for the resource is set to YES.

Inhibited

The resource is not in its desired state and automation is unable to proceed because of a problem with a supporting resource. Operators should try to fix supporting resources with compound status *Problem* and *Denied*.

Awaiting [Automation]

The resource is not in its desired status and SA z/OS is waiting for its supporting resources to reach the appropriate state.

InAuto[mation]

SA z/OS is in the process of starting or stopping the resource.

Degraded

For a group it means that it is partially running, but not at full capacity. For normal resources, it can mean that the resource is Starting or Stopping, or that the application is suffering from a performance or throughput problem (corresponding to automation agent status HALTED).

Satisfactory

The resource's desired and observed statuses are corresponding; no further automation or operator activity is required.

Health Status

With application-specific performance and health monitoring, a separate status shows up to inform you about the application's health.

Possible values are:

Normal

The observed resource is behaving normally.

Warning

The resource's health state has crossed the warning threshold.

Minor The resource's health state has crossed the minor threshold.

Critical

The resource's health state has crossed the critical threshold.

Statuses Supplied by the Automation Manager

- | **Fatal** The resource's health state has crossed the fatal threshold.
- | **Unknown**
- | The health state of the resource is not known.

Statuses Supplied by Automation Agents

There are different types of statuses that may be applied to an application. They are all retrieved on the side of the automation agent. These are:

- "Automation Agent Statuses"
- "Application Monitor Status" on page 45
- "Automatic Restart Manager Statuses" on page 46

Furthermore, this chapter discusses the following related topics:

- "Mapping the Automation Agent Statuses to the Automation Manager Observed Status" on page 47
- "SA z/OS Processes Involved in Status Determination" on page 48

Automation Agent Statuses

SA z/OS defines different automation agent statuses that it applies to automated resources. The automation agent status of a resource is determined from a combination of its application monitor status, its desired status, recent history, and intended action.

These statuses are used on the automation agents to control the automation. They are translated to *observed statuses* (see "Mapping the Automation Agent Statuses to the Automation Manager Observed Status" on page 47) and then sent to the automation manager.

On the automation agent you can use the DISPSTAT and DISPINFO commands to see the automation statuses.

The following is a brief description of each automation agent status.

Value	Meaning
ABENDING	<p>The application is undergoing a recoverable abnormal end. This status is entered when a TERMINATION message with the ABEND=YES attribute is received. It remains in ABENDING until its final termination message is received. If the application is defined to Automatic Restart Manager it may be posted to EXTSTART when Automatic Restart Manager attempts to recover it. If the application has not exceeded its critical threshold, SA z/OS posts the application to RESTART and attempts to restart it. If it has exceeded its critical threshold it is put into BROKEN status when it has been cleaned up, and SA z/OS does not attempt to start it. If it is slow to leave the system after its FINAL message is received it may go to ZOMBIE status.</p> <p>If the application is undergoing a normal shutdown by SA z/OS no further shutdown commands are issued. The shutdown may be resumed by using the INGREQ command with a stronger type.</p>

Statuses Supplied by Automation Agents

Value	Meaning
ACTIVE	<p>The application is running, but is not yet ready for work. An application can be put into ACTIVE status in response to the following conditions:</p> <ul style="list-style-type: none"> (i) SA z/OS has received an ACTIVE message for the application. (ii) The SA z/OS startup checker has run for the application when its automation status was STARTED. The startup checker found the application monitor status for the application to be ACTIVE. (iii) During routine status checking the application monitor status for the application was found to be ACTIVE when its automation status indicated it should be otherwise. (iv) SA z/OS attempted to start the application, but found that its application monitor status was ACTIVE. (v) SA z/OS found that the Automatic Restart Manager status for the application is STARTING, but has not received any messages concerning the application status. (vi) SA z/OS checked an attempt by Automatic Restart Manager to restart the application and found that its Automatic Restart Manager status is UNREGISTERED, but the application monitor status for the application is ACTIVE. <p>An application remains in ACTIVE status until its UP message is received. If the application is starting or restarting, an SA z/OS monitoring check for the application is done after its start delay time. If the application is not found it is put into INACTIVE status. If it is found, but is not yet UP, it is put into STARTED2 status.</p>
ASSIST	<p>This is an unusual status. When an application has ASSIST=DISPLAY coded for one or more of its automation flags, SA z/OS puts the application into ASSIST status whenever it wants to issue a command that is controlled by that automation flag. ASSIST indicates that SA z/OS is ready to take an action which will affect an application, but that you must approve the action before it occurs. An application can enter ASSIST status when SA z/OS is going to start it, stop it, or perform secondary automation for it.</p> <p>When an application is in ASSIST status you can access an ASSIST panel using SDF. This panel is used to tell SA z/OS to issue the command where it should be issued, to issue the command on the your account, or to ignore the command. The regular monitor ignores applications that are in ASSIST status, and most command dialogs will not process them.</p> <p>To get an application out of ASSIST status you must respond to all the ASSIST panels that are present in SDF for the application. If the commands are issued in the correct place, the application should come up. You can use the DISPASST command dialog to see assist settings, and the SETASST command dialog to change them dynamically.</p>
AUTODOWN	<p>The application is shut down. SA z/OS may restart it to comply with an operator request. If the shutdown specified that the application was to be restarted, it is put into RESTART status when the shutdown is complete.</p> <ul style="list-style-type: none"> (i) SA z/OS has shut the application down at the request of an operator. (ii) When SA z/OS initialized, the operator replied NOSTART to the AOF603D WTOR. Any applications which would have been put into the DOWN status during initial status determination have instead been put into the AUTODOWN automation status. <p>You can use the SETSTATE command to change the application status to either RESTART or CTLDOWN.</p> <p>SA z/OS may attempt to restart an application in AUTODOWN status when SA z/OS is reloaded, or when an operator requests SA z/OS to start one of the application descendents. In both cases, the application goes to RESTART status.</p>

States Supplied by Automation Agents

Value	Meaning
AUTOTERM	<p>SA z/OS is in the process of shutting the application down. The shutdown is in response to a INGREQ REQ=STOP command. This status persists until SA z/OS is sure that the application has been cleaned up.</p> <p>Many things may happen to an application that is being shut down. If the shutdown is successful, the application is placed in either AUTODOWN or CTLDOWN status. If the shutdown specified that the application should be restarted it goes through AUTODOWN to RESTART status.</p> <p>If the application abnormally ends while it is being shutdown it may go into either ABENDING or BREAKING. A normal shutdown will stop processing an application that abends, but other shutdowns will continue. If the shutdown runs out of commands to issue, the application is placed into STUCK status. If it has problems shutting down, the application may be placed into ZOMBIE status.</p>
BREAKING	<p>The application is undergoing a nonrecoverable abend; that is, it has received a termination message specifying BREAK=YES. If the application is undergoing a normal shutdown by SA z/OS no further shutdown commands are issued. The shutdown may be resumed by using the INGREQ REQ=STOP command. This status persists until SA z/OS receives its final termination message and is sure that the application has been cleaned up. If the termination experiences difficulties, the application may be posted to ZOMBIE status.</p>
BROKEN	<p>The application has suffered a nonrecoverable abend. SA z/OS will not restart it. An application can be put into BROKEN status in response to the following conditions:</p> <ul style="list-style-type: none"> (i) The application has suffered a nonrecoverable abend, indicated by the reception of a TERMINATION message with the BREAK=YES attribute. (ii) The application has suffered sufficient recoverable abends to exceed its critical threshold. (iii) The application has suffered a prestart or startup failure. <p>This status is preserved across a recycle of SA z/OS or a re-IPL of the processor, unless the application has its Start on IPL/RECYCLE Option set to YES.</p>
CTLDOWN	<p>The application is shut down and SA z/OS is not allowed to restart it.</p> <ul style="list-style-type: none"> (i) An operator asked SA z/OS to shut the application down and not to restart it until authorized to do so by an operator. (ii) An operator used a SETSTATE command to tell SA z/OS that an application should not be restarted until an operator authorizes SA z/OS to do so. (iii) SA z/OS was recycled, the application monitor status is INACTIVE, and its Start on RECYCLE Option is NO. (iv) SA z/OS was started for the first time, the application monitor status is INACTIVE and its Start on IPL Option is NO. (v) SA z/OS was recycled while the application was in a CTLDOWN status. If the AOFCTLOPT advanced automation option is set then the application status remains CTLDOWN regardless of all other settings, parameters, and options, providing its application monitor status remains INACTIVE. <p>You can use the SETSTATE command to change an application status from CTLDOWN to RESTART or AUTODOWN, in which case SA z/OS will attempt to restart it.</p>

States Supplied by Automation Agents

Value	Meaning
DOWN	<p>The application has not been started during the lifetime of this SA z/OS.</p> <p>The DOWN status is set only during initial status determination and is possible only if the application monitor status is INACTIVE. The automation status of the application when SA z/OS was last shut down on this system is used in the following manner to determine if it is to be placed into the DOWN status.</p> <ul style="list-style-type: none"> (i) The previous automation status was not one of STOPPED, CTLDOWN or BROKEN. (ii) The START on IPL option for the application is YES and this is the first time that SA z/OS has been started since MVS was last IPLed. If the previous automation status was CTLDOWN and the AOFCTLOPT advanced automation option is set then the automation status remains as CTLDOWN. (iii) The START on RECYCLE option for the application is YES and this is not the first time that SA z/OS has been started since MVS was last IPLed. If the previous automation status was CTLDOWN and the AOFCTLOPT advanced automation option is set, then the automation status remains as CTLDOWN. (iv) The application does not have a Start on IPL Option of NO. If it does, its status is set to CTLDOWN. (v) The application does not have a Start on RECYCLE Option of NO. If it does, its status is set to CTLDOWN.
ENDED	<p>This status is used for transient applications only, and indicates that the job for the application has finished and left the system without any errors. Any start-dependent resources for the application will be started as though it were a normal z/OS subsystem that was UP.</p> <p>If the transient application can be rerun, you can use the SETSTATE command to restart it. If the transient application cannot be rerun, it will remain in ENDED status.</p>
ENDING	<p>A transient application is in the process of terminating. A transient application goes to ENDING status when a termination message is received for it, and it is not being shut down by SA z/OS. This status shows that the application is terminating, but that this is expected. A transient application may also go to ENDING if an operator is shutting it down outside SA z/OS control, or if it has abnormally ended, but the abend messages are being treated as normal termination messages.</p> <p>The application remains in ENDING status until either:</p> <ul style="list-style-type: none"> • an abend message is received, which will put it into either ABENDING or BREAKING status. If the application abends then either SA z/OS or Automatic Restart Manager can restart it. • the application final termination message is received, at which point the RESTARTOPT for the application is checked. If it is ALWAYS then the application is put into RESTART status and SA z/OS will attempt to restart it. If it is anything else, the application go to ENDED status. It is assumed that if a transient application ends normally then it will deregister from Automatic Restart Manager. If it is slow to clear the system after its FINAL message is received it may go to ZOMBIE status. <p>If an ACTIVE or UP message is received for the application, its automation status is changed to either ACTIVE or UP, as appropriate.</p>
EXTSTART	<p>SA z/OS has determined that the application is being started or restarted by an agent external to SA z/OS. In situations where SA z/OS is able to identify the external agent (such as Automatic Restart Manager), it takes appropriate steps to monitor that agent's actions and, if necessary, step in to assist it. An application can be put into EXTSTART status in response to the following conditions:</p> <ul style="list-style-type: none"> (i) SA z/OS is unable to identify the external agent. (ii) Automatic Restart Manager is in the process of restarting the application.

Statuses Supplied by Automation Agents

Value	Meaning
FALLBACK	<p>The application is not running on the primary system where it should run, but this status has been encountered for this application on one of its secondary systems. It should be active on another system. If the other system fails, the application can fall back to this system, where it could possibly be restarted. However, SA z/OS, will not perform the restart on the fallback system, but this may be done by an operator request. This is implemented to leave the decision of restarting the application on the fallback system to the operator.</p> <p>An application can be put into FALLBACK status in response to the following conditions:</p> <ul style="list-style-type: none"> (i) It is defined with a secondary association on this system. (ii) An operator has used the SETSTATE command to put the application into the MOVED status. If this is one of the secondary systems for the application it will go to FALLBACK instead.
HALFDOWN	<p>SA z/OS was in the process of shutting the application down, but the stop request was canceled while it was in progress. The application shutdown did not complete (for example, ASCBs may still be active). You may sometimes find that some, but not all, of the shutdown commands have been issued. To recover an application from HALFDOWN status you must determine where it is in its shutdown process and complete the shutdown manually. Applications go into HALFDOWN only when you cancel a stop request. Alternatively, you can use SETSTATE to put the application back into UP or RUNNING status.</p>
HALTED	<p>The application is still running, but something has happened which may have severely impacted its capabilities.</p> <ul style="list-style-type: none"> (i) SA z/OS has received a HALT message for the application. (ii) SA z/OS has detected that the application represents a JES2 application which is running short of spool space. (iii) The Automatic Restart Manager status for the application is ELSEWHERE, but SA z/OS found its application monitor status to be either STARTING or ACTIVE. <p>An application is taken out of HALTED status if its UP message is received. Also, operators may use the SETSTATE command to put the application into UP/RUNNING status.</p>
INACTIVE	<p>At some point in monitoring, the application monitor status was INACTIVE when the automation status indicated that it should be either ACTIVE or STARTING. An application can be put into INACTIVE status in response to the following condition:</p> <ul style="list-style-type: none"> (i) The application monitor routine found the application's status to be INACTIVE. <p>If the application RESTART option is ALWAYS, the application is restarted rather than put into INACTIVE status.</p>
MOVED	<p>The application is not running; this is one of its primary systems: it should be active on another system. An application can be put into MOVED status in response to the following condition:</p> <ul style="list-style-type: none"> (i) An operator has used the SETSTATE command to put the application into the MOVED status. This is possible only on a primary system.

Statuses Supplied by Automation Agents

Value	Meaning
RESTART	<p>The application is ready to be started. It has been previously active in the system. An application can be put into RESTART status in response to the following conditions:</p> <ul style="list-style-type: none"> (i) The application abended and, after checking thresholds, SA z/OS is allowed to restart it. (ii) SA z/OS has shut the application down in response to an operator request and is now preparing to restart it. (iii) An operator has used the INGREQ REQ=START command to ask SA z/OS to restart the application. (iv) SA z/OS checked an attempt by Automatic Restart Manager to restart the application and found that its Automatic Restart Manager status is UNREGISTERED and the application monitor status for the application is INACTIVE. This implies that the attempt by Automatic Restart Manager to restart the application timed out while the application was in RESTARTING status. SA z/OS changes the automation status of the application to RESTART and attempts to start the application itself. <p>During restart processing, the application RESTART automation flag is checked. If it is turned on, the application start commands are issued and the application is put into STARTED status. If the RESTART automation flag is off, the application remains in RESTART status and the startup monitor cycle initiates the startup process each time it runs.</p>
RUNNING	<p>This status is equivalent to UP, but is used for transient applications. It indicates that the UP message has been received for the transient application, or an operator has used the SETSTATE command to change the status of a transient application to RUNNING. A transient application is one which SA z/OS expects to terminate on its own. When the job finishes the application goes through ENDING status to ENDED, at which point its descendants are started. Unlike the UP status, the descendants of a transient application are not started until it has terminated.</p> <p>A transient application should leave RUNNING status on its own. If it gets stuck, you should use investigate it. You can use the INGREQ REQ=STOP command to put it into AUTODOWN status.</p>
STARTED	<p>The commands to start the application have been issued, but it has yet to start running. An application can be put into STARTED status in response to the following conditions:</p> <ul style="list-style-type: none"> (i) SA z/OS has issued, or will soon issue, the commands. (ii) When SA z/OS attempted to start the application it found that the application monitor status for the application was STARTING. (iii) During initial status determination, SA z/OS found that the application monitor status for the application was STARTING. (iv) SA z/OS checked an attempt by Automatic Restart Manager to restart the application and found that its Automatic Restart Manager status is UNREGISTERED, but the application monitor status for the application is STARTING. <p>Note that the relevant automation flag, Initstart or Restart, must be on. The application startup commands as defined in the automation control file are issued after the application is placed in STARTING status.</p> <p>An application remains in STARTING status until either its ACTIVE or its UP message arrives. After the application start delay time, an SA z/OS monitoring check is issued for it. If it is not found, it is put into INACTIVE status. If it is found, but is not yet UP, it is put into ACTIVE status.</p>

Statuses Supplied by Automation Agents

Value	Meaning
STARTED2	<p>The application has become active, but has not indicated that it is ready to accept work within its start delay. An application can be put into STARTED2 status in response to the following conditions:</p> <ul style="list-style-type: none"> (i) The startup checker found that the application monitor status was still STARTING. (ii) The startup checker was called for an application whose automation status was ACTIVE. (iii) SA z/OS found the Automatic Restart Manager status for the application to be AVAILABLE-TO. (iv) SA z/OS checked an attempt by Automatic Restart Manager to restart the application and found that its Automatic Restart Manager status is one of AVAILABLE-TO, RESTARTING or RECOVERING. (v) SA z/OS checked an attempt by Automatic Restart Manager to restart the application and could not find a better status to put it into. Its application monitor status is ACTIVE. This covers situations where the restart checker is unable to determine the Automatic Restart Manager status for the application, or its status is FAILED. <p>An application remains in STARTED2 status until either its UP message arrives or an operator uses the SETSTATE command to change the application status to UP/RUNNING.</p>
STOPPED	<p>The application has been shut down by an external agent, such as an operator cancel. SA z/OS is not permitted to restart it and will not allow Automatic Restart Manager to restart it. This status is preserved across a recycle of SA z/OS or a re-IPL of the processor unless the application has its Start on IPL/RECYCLE Option set to YES.</p> <p>An application remains in STOPPED status until an operator uses the SETSTATE command to change its status to either RESTART or CTLDOWN.</p> <p>An application may also leave STOPPED status if it is restarted outside the control of SA z/OS. In this case, it goes to either the ACTIVE or the UP status, depending on which message is received.</p>
STOPPING	<p>The application is being shut down by an external agent. This status is entered when a TERMINATION message is received while SA z/OS is not in the process of shutting the application down. It may also indicate that the application is abending, but theabend messages issued are being treated as normal termination messages by automation.</p> <p>An application that is STOPPING remains in that status until either anabend message is received which puts it into ABENDING or BREAKING status, or the final termination message for the application is received, at which point the application RESTART option is checked. If RESTART is ALWAYS, the application is put into RESTART status and SA z/OS attempts to restart it. Otherwise, the application goes to STOPPED status. If it is slow to clear the system after the final message is received, it may be placed into ZOMBIE status.</p> <p>If an ACTIVE or UP message is received for the application, its automation status is changed to either ACTIVE or UP, as appropriate.</p> <p>Automatic Restart Manager interaction with this status depends on a number of factors. If the application is deregistered then there is no interaction. If the application is registered then Automatic Restart Manager will attempt to restart it. If the application goes to STOPPED status before the SA z/OS/Automatic Restart Manager exit is invoked then SA z/OS will tell Automatic Restart Manager not to restart the application. If it does not, then SA z/OS will tell Automatic Restart Manager that it can restart the application.</p>
STUCK	<p>An application can get STUCK when it is being shut down. This is because SA z/OS has run out of NORM, IMMED, or FORCE shutdown commands.</p>

Value	Meaning
UP	<p>The application has finished initializing and is ready for work. An application can be put into UP status in response to the following conditions:</p> <ul style="list-style-type: none"> (i) SA z/OS has received an UP message for the application. (ii) An operator has used the SETSTATE command to change the application automation status. In this case, SA z/OS assumes that the operator has ensured that the application is actually UP. (iii) During initial status determination the application monitor status was found to be ACTIVE. (iv) SA z/OS found the Automatic Restart Manager status for the application to be AVAILABLE. <p>There are a number of ways for an application to leave the UP status:</p> <ul style="list-style-type: none"> • If it is shutdown with the INGREQ REQ=STOP command, it goes to AUTOTERM status • If it is shutdown outside SA z/OS, it goes to STOPPING status • If it abends, it might go to STOPPING, ABENDING or BREAKING status • If it has problems, it may go to HALTED status • If the regular monitor cannot find it, it will go to INACTIVE status • If the application abends, SA z/OS does not pick up the abend messages, and Automatic Restart Manager detects that the address space has ended, the application may go to EXTSTART.
ZOMBIE	<p>When an application is leaving the system it can get into ZOMBIE status. This indicates that the final termination message for the application has been received but that SA z/OS monitoring in TERMMSG still finds the application. The application is put into ZOMBIE status if this situation persists for more than twice the application termination delay time.</p> <p>There are three ways in which an application can get into ZOMBIE status:</p> <ul style="list-style-type: none"> • If MVS is slow in clearing the application and the termination delay time is short. • If there are two jobs with the same name in the system, one of which is the application. When either of them terminates and SA z/OS does not know the address space id of the application or does not get the address space id with the termination message, SA z/OS assumes that the application has stopped, but SA z/OS monitoring will find the other job. To change the status to UP, either manually shut down the other job, or use the SETSTATE command to change the application status back to UP. • The job may have become stuck in the system after issuing its final message. <p>From ZOMBIE status, if the application suffers an unrecoverable abend it will go into BREAKING status.</p> <p>Note: The START ON IPL and START ON RECYCLE options of the customization dialog may override these resource statuses at SA z/OS IPL or RECYCLE, resulting in SA z/OS starting the subsystem. If either START ON IPL or START ON RECYCLE are set to NO, the application is placed in CTLDOWN status.</p>

Application Monitor Status

SA z/OS determines the application monitor status of an application by running a routine that you specified in the customization dialog. You can specify the routine for an individual application, and you can specify a default application monitor routine for applications on an entire system. For further information on application monitor routines, refer to *System Automation for z/OS Defining Automation Policy*.

Statuses Supplied by Automation Agents

Table 6. Regular Monitoring: Automation Status Determination

Application Monitor Status	Automation Status Change
ACTIVE	The application automation status is changed to ACTIVE.
STARTING	The application automation status is changed to STARTED and a startup checker is scheduled for it.
INACTIVE	If the application is defined with a RESTART option of ALWAYS, the automation status is changed to RESTART and the restart process is invoked. If its RESTART Option is not ALWAYS, its automation status is changed to INACTIVE.

Automatic Restart Manager Statuses

The z/OS Automatic Restart Manager statuses are determined by SA z/OS and are used to help determine automation statuses. These statuses are included here as background information so that you can better understand the SA z/OS automation statuses. They are a superset of the statuses that Automatic Restart Manager uses, the added statuses being ELSEWHERE and UNREGISTERED. Possible values for the Automatic Restart Manager status are:

Value	Meaning
AVAILABLE	The element name is registered with Automatic Restart Manager, the job is running on this system and the Automatic Restart Manager status for it is AVAILABLE.
AVAILABLE-TO	The element name is registered with Automatic Restart Manager and the job is running on this system. The job has been restarted by Automatic Restart Manager, has issued its IXCARM REGISTER call, but did not issue its IXCARM READY call within the ready timeout defined to z/OS for it.
STARTING	The element name is registered to Automatic Restart Manager and the job is running on this system. The job was started by an agent external to MVS, has issued its IXCARM REGISTER call, but has not yet issued its IXCARM READY call. Unlike the RECOVERING status, this status does not time out to the AVAILABLE-TO status.
RECOVERING	The element name is registered to Automatic Restart Manager and the job is running on this system. Automatic Restart Manager has issued the restart command for the application, it has issued its IXCARM REGISTER call, but has not yet issued its IXCARM READY call. If it takes too long to do this, it is placed in the AVAILABLE-TO status.
FAILED	The element name is registered with Automatic Restart Manager, the job has failed, but was last active on this system. It is unknown at this point if Automatic Restart Manager will attempt to restart the job. SA z/OS will continue to report an elements status as FAILED until MVS either deregisters it, or is committed to restarting it. MVS query commands issued during this time will probably return a RESTARTING status, however MVS Automatic Restart Manager is still consulting its policies and exits to determine whether or not it should restart the element.

Statuses Supplied by Automation Agents

Value	Meaning
RESTARTING	The element name is registered with Automatic Restart Manager, the job has failed and Automatic Restart Manager is attempting to restart it. This status is entered very early in the restart process, before Automatic Restart Manager is committed to restarting the element. SA z/OS continues to report the status as FAILED until Automatic Restart Manager <i>is</i> committed to issuing the restart commands. This means that MVS query commands may report a status of RESTARTING while SA z/OS reports a status of FAILED. The RESTARTING status is exited when the job issues its IXCARM REGISTER call. If it takes too long to do this, its element name is deregistered.
ELSEWHERE	The element name is registered with Automatic Restart Manager, but is currently active on a different system. Unless that system fails, Automatic Restart Manager will not move the application to this system.
UNREGISTERED	The element name is not, currently, registered with Automatic Restart Manager. This may mean that the application has never been active within the sysplex, or it may mean that it failed and Automatic Restart Manager could not recover it. If the SA z/OS routine to determine an Automatic Restart Manager status finds that Automatic Restart Manager recovery is inactive, the UNREGISTERED Automatic Restart Manager status is returned for all enquiries. This means that SA z/OS effectively ignores Automatic Restart Manager.

Mapping the Automation Agent Statuses to the Automation Manager Observed Status

Table 7 shows how the current resource statuses retrieved by the related automation agents are mapped to the observed status used by the automation manager:

Table 7. Automation Agent to Automation Manager Status Mapping

Automation Agent	Observed	Automation	Notes
DOWN	SoftDown	Idle	The automation agent is ready to perform an initial start of the resource.
RESTART	SoftDown	Idle	The automation agent is ready to perform a startup of the resource.
STARTED	Starting	Busy	The automation agent has issued the start commands for the resource.
EXTSTART	Starting	Busy	This means that the automation agent has detected that an external agency (probably ARM) is in the process of starting the resource.
ACTIVE	Starting	Busy	The automation agent has received some indication that the start is successful, although it is not yet complete.
STARTED2	Problem	Problem	Although the resource is still active, it does not appear to have started properly within the time period that the automation agent expected it to start within.
UP	Available	Idle	The resource has entered a stable state and is available for usage.
HALTED	Degraded	Busy	The resource has encountered a problem, but is still active. The automation agent may be attempting to return it to an UP state.

Statuses Supplied by Automation Agents

Table 7. Automation Agent to Automation Manager Status Mapping (continued)

Automation Agent	Observed	Automation	Notes
RUNNING	Starting	Busy	The resource is a transient subsystem that is currently executing.
ENDING	Starting	Busy	The resource is a transient subsystems that is terminating.
AUTOTERM	Stopping	Busy	The automation agent is in the process of stopping the resource.
STOPPING	Stopping	Busy	The automation agent has detected that the application is terminating normally outside of its control.
ABENDING	Stopping	Busy	The automation agent has determined that the application is terminating abnormally but can be recovered.
BREAKING	Stopping	Busy	The automation agent has determined that the resource is terminating abnormally in a nonrecoverable fashion.
ENDED	Available	Idle	The resource is a transient subsystem that has completed successfully.
AUTODOWN	SoftDown	Idle	The resource is down and may be restarted as needed by automation.
CTLDOWN	HardDown	Idle	The resource is down and may not be restarted by automation.
STOPPED	HardDown	Idle	The automation agent has determined that the resource has been stopped normally outside the control of automation and its policy indicates that an automated restart is not allowed.
BROKEN	HardDown	Idle	The automation agent has determined that the application has failed in a nonrecoverable fashion.
INACTIVE	SoftDown	Idle	The automation agent has found the resource to be 'missing'. It may be restarted by automation.
HALFDOWN	Degraded	Problem	A shutdown for the resource by the automation agent was cancelled at the users request. Although the resource is still active, it's status is indeterminate.
STUCK	Stopping	Problem	While trying to stop or start the resource, the automation agent ran out of instructions.
ASSIST	-	-	Not posted.
ZOMBIE	Problem	Problem	The automation agent believes that the resource should be inactive, but it is not.
MOVED	Standby	Idle	The automation agent has determined that the resource is active elsewhere within the sysplex. The manager may reactivate it on this system.
FALLBACK	Standby	Idle	The resource is a backup for a resource which normally runs on a different system within the sysplex. The manager may start it if it needs to.

SA z/OS Processes Involved in Status Determination

This section describes a number of SA z/OS processes that are involved in determining the various statuses of resources. A resource's status is determined by the automation agent and reported to the automation manager.

Initial Status Determination

This process occurs at the end of SA z/OS initialization and after a

configuration refresh. It attempts to determine a suitable automation status for each application that is defined to SA z/OS.

Subsystem Startup

This process is invoked whenever SA z/OS is about to start an application. It checks that conditions are right for the application to be started and that SA z/OS is allowed to start the application. If appropriate, the subsystem startup process then issues the commands to start the application and changes the application automation status to STARTED.

Startup Checker

This process is run for each application after SA z/OS has issued the startup commands, after the amount of time specified by the Start Timeout defined for the application. It changes the automation status from STARTED to STARTED2 if the application fails to start within its expected start delay.

Regular Monitoring

This process runs at an interval defined by the automation programmer in the customization dialog. Normally the time interval is between 30 minutes and 12 hours. The purpose of regular monitoring is to carry out a reality check to ensure that the current SA z/OS automation status for an application is consistent with the current application status as determined by the application monitor routine.

To help determine consistency, SA z/OS statuses are divided into four groups:

UP STATUSES

These statuses mean that the application monitor status should be ACTIVE. They include UP, RUNNING, ACTIVE, and HALTED.

DOWN STATUSES

These statuses indicate that the application monitor status should be INACTIVE. They include DOWN, RESTART, INACTIVE, AUTODOWN, CTLDOWN, BROKEN, STOPPED, ENDED, FALLBACK, and MOVED.

STARTING STATUSES

These statuses indicate that the application monitor status may be STARTING or ACTIVE. They include STARTED, STARTED2, and EXTSTART.

TRANSITIONAL STATUSES

These statuses are either a part of the shutdown process, or indicate that the application may or may not be present in the machine. ALL application monitor statuses are possible with any transitional status. They include AUTOTERM, STUCK, ZOMBIE, BREAKING, ABENDING, HALFDOWN, STOPPING, ENDING, and ASSIST.

If regular monitoring finds a status mismatch, it changes the application automation status according to Table 6 on page 46.

It should be noted that several other routines have similar status mismatch detection capabilities built in.

Secondary Monitor

This is invoked, after a small delay, from the application monitor routine. This process performs a recheck for a specific application that the

Statuses Supplied by Automation Agents

application monitor routine found to be in an unusual (generally a transient) state. It is responsible for making some of the transitions outlined under the Routine Monitor process.

Shutdown

This process is normally invoked by an operator to shut down one or more applications. The shutdown process is responsible for the sequencing of the various shutdown commands. Depending on the parameters it is invoked with, the application that is shut down may be put into an AUTODOWN or CTLDOWN automation status, or it may be restarted.

Resynchronization Process (Environmental Analysis Process)

It may be invoked manually by an operator using the RESYNC command. This process assesses the current status of the various parts of the z/OS system that SA z/OS automates. It issues display commands to obtain information on WTORs and sysplex resources. It also invokes the Initial Status Determination process. When it runs, all previous status information is discarded.

Using SA z/OS for Monitoring

As an operator, you can monitor and display information about the resources using two types of interfaces:

- The NMC workstation provides monitoring capability for system operations and processor operations.
- The SA z/OS 3270-type interface
 - provides monitoring capability for processor operations
 - provides monitoring capability for system operations via Status Display Facility (SDF), displaying status information (INGLIST) and detailed resource information (INGINFO and DISPINFO for the automation agent view)

You can also monitor messages and alerts. More information about this topic is given in “How to Monitor Alerts on a NetView Console” on page 133.

If SA z/OS has been configured properly, status mismatches between the actual and the monitored status should not occur. In the customization dialog, the automation programmer can specify a monitoring interval for every resource individually. If such an individual interval is not defined for a resource, then every half hour (which is the default, but the interval is user specifiable) SA z/OS will run a monitor routine to check the resources' status as known to SA z/OS versus its actual status. If the result indicates an error, its automation status will be updated.

In the customization dialog, the automation programmer can define either an individual monitor routine per application or a default monitor routine for a system that will monitor all applications running on that system. You can manually invoke an application's related monitor routine through the MONITOR command. You can use a wildcard for the applications (subsystems) that you want to be monitored, for example, use *abc* for all subsystems containing the string 'abc' in their names.

Resynchronization

If you believe that a lot of the status information provided by SA z/OS is badly out of sync with reality, you can request that SA z/OS reruns its complete subsystem resynchronization process.

This is achieved with the RESYNC command, but it should be used with care. The resynchronization can take a few minutes and during this time a lot of SA z/OS's commands are not available.

Using Group Start and Stop Types

SA z/OS provides the ability to set START and STOP types (and parameters) on groups, as well as on individual resources. These are designed to be used when you are operating the group as a single entity. If you wish to manipulate the individual members of a group, you are better off setting their parameters directly because using the groups as a shortcut could result in undesired behavior. For example, if you wish to stop an application and then restart it with a specific START type, you could use the commands to set the start type for the group, stop the group and then restart it. The resources behind the application that get started will be brought up with the specified start type, however they may not be the resources that were stopped when the group was brought down. If, on the other hand, you wish to stop a resource on a specific system and bring it back up with a specific start type, you should operate directly upon the resource.

Resources

A resource will consume its start type (and parameters) when it is started and its stop type (and parameters) when it is stopped.

Basic Groups

Basic Group -- Start Type

A Basic group will consume its start type once all of its members have consumed theirs and the group has become Available or WasAvailable, provided that the group is desired Available.

basic group start type

What can go wrong?: As with the start process, if something prevents a group from becoming inactive once its shutdown has been started, the group's stop type will not have been consumed, even though it may have been consumed for some or all of the members.

If a member of the group abends or is manually stopped, its stop type will not have been consumed, and so the group will not consume its stop type.

Basic Group -- Stop Type

A Basic group will reset its stop type once all of its members are SoftDown, HardDown, Unknown, SysGone or Standby.

What can go wrong?: As with the start process, if something prevents a group from becoming inactive once its shutdown has been started, the group's stop type will not have been consumed, even though it may have been consumed for some or all of the members.

If a member of the group abends or is manually stopped, its stop type will not have been consumed, and so the group will not consume its stop type.

Move and Server Groups

When Move and Server Groups consume a start or stop type, they also consume the start and stop types and parameters from all of their members. This means that if you set a start type on a server group, stop it and then start it, the two systems where the resources for it are brought up will use the start type -- however it will be wiped before it is used on the other systems. If you want to set a 'next-time-it's-started-on-this-system' start type, you should set it for the individual resources, not for the group.

Move and Server Group -- Start Type

A Move or Server group will consume its start type once the group becomes Available or WasAvailable after having been SoftDown (that is, the group has been started with the start type set), provided that the group is desired Available

What can go wrong?: If the group never becomes available, some members may have consumed their start type, but the group will not have consumed its start type.

The detection is based upon the group's observed state which, in the case of a server group, is set from the SATISFACTORY target, not the AVAILABILITY target. If something causes a delay in the startup of some members, it is possible the start type will only be applied to the first systems to be started (up until the Satisfactory target is met), and will not be applied to additional systems started to bring the group up to its Availability target.

You may get unpredictable behavior trying to set the Start type for a group with an Availability target of 0. The observed state of such a group is based solely upon its desired state. There is no guarantee that a start type would be used for any of its members before it was consumed.

If a member is manually started its start type will not have been used, however when the group becomes Available, it will be consumed.

Move and Server Group -- Stop Type

A Move or Server group will consume its stop type once the group enters a down state (SoftDown, HardDown, Unknown, SysGone or Standby) after having been Available or WasAvailable (that is, after the group has been stopped with the stop type set).

What can go wrong?: If the group never becomes unavailable, some (or all) members may have consumed their stop type, but the group will not have consumed its stop type.

If a member abends or is manually stopped, it will not have been stopped with the correct stop type -- however the group will consume its stop type and the stop type of the abended or stopped member when the other members of the group come down.

Chapter 3. Starting and Stopping SA z/OS

This chapter contains the following sections:

- “Starting, Stopping, and Maintaining the Automation Manager”
- “Starting and Stopping I/O Operations” on page 57
- “Starting NetView (System Operations)” on page 57
- “Starting and Stopping Processor Operations” on page 58

Starting, Stopping, and Maintaining the Automation Manager

Normally, user interaction with the automation manager is limited to using INGXXXX commands, for example, to send it requests to start or stop resources. In special cases, however, there may be a need to interact with the automation manager during the initial start process and when it is necessary to stop an automation manager instance.

Starting the Automation Manager—START (S)

The automation manager runs in a z/OS address space of its own. Normally, you would add this start command to the COMMNDxx PARMLIB member so that the automation manager is automatically started at IPL time as described in *System Automation for z/OS Planning and Installation*. Alternatively, it may be started by an MVS start command that calls a module that initializes the automation manager.

In certain cases, you can also use the START (S) command.

Because the automation manager runs as a started task, the start command format may be modified by the installation. However a sample procedure is provided and can be used in most installations without modification. In addition to the various data sets identified by this procedure, there are several parameter values that may be specified. None of these parameters are required due to default values and/or additional sources of the information that are provided. You can find a sample startup procedure called INGEAMSA in the SINGSAMP sample library, see also *System Automation for z/OS Planning and Installation*.

In most cases, the parameters specified via the start command, if provided, override information obtained from the automation manager’s PARMLIB member or other default value. The only exception to this is that START=HOT will be ignored when another automation manager instance has claimed primary mode.

Note:

Do not start two instances of the automation manager with the same job name on the same system. They would come up as the primary and secondary automation manager. However, if you issue a stop command with the same job name, it would be accepted by both instances.

Also you cannot start more than nine instances of an automation manager per z/OS system within the sysplex or per single system.

To allow your installation to specify a preference as to which automation manager instance should assume PAM responsibility at sysplex IPL, each automation manager enters a DELAY state. The duration of this state is set with the DELAY parameter of the automation manager initialization member of the PARMLIB (see also the description of the HASPRM00 sample member in *System Automation for z/OS Planning and Installation*). This allows other instances with a lower delay value to assume PAM mode if started within the difference (in seconds) between the respective delay values. This makes sense when you would prefer that the primary automation manager runs on a special powerful processor.

Stopping the Automation Manager—STOP (P)

To allow an automation manager to be stopped or otherwise influenced by an operator, the automation manager "listens" for MVS MODIFY and STOP commands (see also "Making an Automation Manager the Primary One.") When an MVS STOP (P *jobname*) command is received, the automation manager will be terminated. If the automation manager was the primary one, all work in process will be completed prior to actual termination.

If you try to stop the last primary automation manager, you will receive the reply HSAM1390E to which you must confirm the automation manager stop request.

Making an Automation Manager the Primary One

You can use the GO subcommand of the MVS MODIFY (F) command to cancel the initial DELAY invoked when an automation manager instance is started thereby making it eligible to become the primary instance without having to wait for the delay time to complete. This is useful when it is known that a preferred primary system will not become available during a sysplex wide IPL:

```
F jobname,GO
```

Stopping the Automation Manager with Different Options

The STOP subcommand of the MVS MODIFY (F) command may be used to stop an automation manager instance in different ways:

```
F jobname ,STOP,[ NORMAL | IMMED | FORCE | DEFER ],[ NORESTART | RESTART ]
```

The available options are:

NORMAL

The instance will be stopped after all in progress work has been completed

Starting, Stopping and Maintaining the Automation Manager

and committed. No new work will be accepted from the automation agents, however any internally queued work will be processed. This is the default option.

IMMED

The instance will be stopped after the current work items have been completed and committed. Internally queued work may be lost.

FORCE

The instance will be stopped without waiting for in progress work to complete.

DEFER

The instance will be conditionally and normally stopped after the STOPDELAY interval has expired. Termination will occur only if another (secondary) instance is available, or if no automation agent is active.

On a secondary automation manager instance, the above variations are effectively treated as NORMAL as there is no work to be completed by SA z/OS. The second option specifies whether ARM restart processing should be attempted following the completion of the stop request.

NORESTART

The instance will be stopped as requested and no restart attempt will be made by ARM. This is the default option.

RESTART

The instance will be stopped in such a way as to cause ARM to attempt an automatic restart. Note that this will only occur if the instance is properly defined to ARM.

Getting Automation Manager Diagnostic Information

There are two methods of getting diagnostic information from the automation manager activities in case problems occur:

- You can get a snapshot of the state image queue written into a specified data set using the command:
F jobname, DIAGINFO,SNAP,data_set_name
- You can get a log of the automation manager's activities (maybe the ones to reproduce the problem) for problem reporting. You issue a command to start the log which is written to a specified data set. With a second command, you terminate the log:
 - to start recording:
F jobname , DIAGINFO,RECORD,data_set_name
 - to stop recording:
F jobname , DIAGINFO,ENDREC

Note: The DIAGINFO command accepts every data set name without checking if the data set exists or without checking if it is accessed by some other user. In case of conflicts, the automation manager then does not write any diagnostic information, but will not issue a message to the user.

Takeover File Repair

The takeover file that was introduced with SA OS/390 2.1 is a shared VSAM file. The takeover file is allocated to the primary automation manager (PAM) and all secondary managers (SAM) via a DD statement in the startup procedure with a disposition of SHR.

Starting, Stopping and Maintaining the Automation Manager

If a VSAM open error occurs for example due to locking or any other I/O error while performing a hot takeover, the automation manager continues with a warm start rather than a hot start without giving the operator the ability to correct the error situation.

If an error occurs while writing to the takeover file, the automation manager continues to run using the in-storage version of the resources objects but disables updating of the takeover file. The in-storage state information of the resources is not written to the takeover file when the automation manager terminates. The user has lost the ability to continue with a hot start when switching the primary automation manager.

An explicit request to enable the takeover file will retry the "old broken" VSAM file only.

If an error occurs while reading the takeover file, the automation manager disables the takeover file and performs a warm start.

The automation manager is changed to allow the user to repair the takeover file while the PAM as well the SAMs are running. Furthermore the operator is prompted before a hot start is degraded to a warm start due to a takeover file open/read error:

1. Read initial record (event handler)

Read all records while processing the ReadConfig request

Dynamic Takeover File

Starting with SA z/OS, the takeover file is dynamically allocated during the initialization of the primary automation manager (PAM). The old DD-statement HSATKOVVR in the startup procedure is no longer used and must be removed.

An automation manager started as SAM does not allocate the takeover file.

During initialization of the automation manager, the name of the takeover file is read from HSAPRMxx and the takeover file is allocated using the same DD statement as used before (HSATKOVVR). The name of the takeover file cannot be changed when one automation manager (PAM or SAM) is running. All automation managers (in the same XCF group) must use the same takeover file name.

In case of a hot start or takeover, and the takeover file allocation or a VSAM open/read error occurs, the operator will be prompted to make a decision:

1. Retry reading the takeover file
2. Continue with a warm start

If the operator continues with a warm start and this fails, the automation manager terminates and causes a takeover.

If the operator continues with a hot start and the previous allocation or a VSAM open/read error reoccurs the automation manager prompts the user again.

In case of cold or warm start the takeover file will be allocated during automation manager initialization. If it cannot be allocated the automation manager terminates. No WTOR is issued because this is not necessary for cold or warm start.

Note

1. All automation managers in the same XCF group must run SA z/OS before the takeover file can be removed from the primary automation manager in-flight for repair. This is because the downlevel automation manager(s), including SAMs, have the takeover file allocated statically in shared mode.
2. If the Automatic Restart Manager (ARM) is active, it triggers the automation manager restart. This then uses the same start parameters as the original start and thus the automation manager performs a *hot* start again.

Starting NetView (System Operations)

NetView must be started on the host before you can use the system operations and processor operations of SA z/OS. NetView can be automatically initialized when you start z/OS. For general information on initializing NetView, see *Tivoli NetView for z/OS Installation: Configuring Additional Components*. You may have a dual NetView environment: a Networking NetView (which runs NetView Graphic Monitor Facility and other networking facilities) and an Automation NetView (which runs SA z/OS). This topic assumes that both instances of NetView are initialized at the focal point system and that you are logged on to a NetView user ID at the SA z/OS workstation.

The system programmer may have set up your system so that SA z/OS is started automatically when the Automation NetView initializes. (Directions for performing this setup are given in *System Automation for z/OS Planning and Installation*). As SA z/OS initializes, a multiline write-to-operator (MLWTO) message, AOF767I, is issued describing the options available and system default actions. See Appendix A, "Automation Initialization Options," on page 259 for the options that can be defined and for details about the MLWTO. To continue automation, enter *R number*

(where *number* is the message reply number of the accompanying AOF603D message). This is the default. If you do not issue a reply within two minutes, automation continues and SA z/OS loads automation policy from an automation control file.

Note: Using the AOFINITREPLY variable you can suppress the issuing of the AOF603D message and have SA z/OS initialize with the default settings (see also *System Automation for z/OS Customizing and Programming*).

Starting and Stopping I/O Operations

It is likely that the system programmer will have customized SA z/OS to initialize all of its components (system operations, processor operations and I/O operations) and related applications automatically. If this is not the case, you need to start the SA z/OS I/O operations manually.

1. I/O operations is started as an application running on the Automation NetView at the focal point system and at the target systems. To start I/O operations, issue the following command from the z/OS operator console, specifying the I/O operations procedure name that has been created as a member in your SYS1.PROCLIB or USER.PROCLIB (the location of the I/O operations

Starting and Stopping I/O Operations

procedure depends on your installation). In the following example, INGEIO is the procedure name. Optionally, you can append an MVS identifier, which is IO in the example.

```
/S INGEIO.IO,SUB=MSTR
```

You can omit SUB=MSTR, but this is not recommended. If SUB=MSTR is omitted and the JES subsystem is not running, I/O operations will neither start nor run.

2. You stop I/O operations by issuing one of the commands **/P INGEIO** or **/P IO**.

Starting and Stopping Processor Operations

1. Processor operations is started as an application running on the automation NetView at the focal point system. To start processor operations, issue the following command from a NetView operator console:

```
ISQSTART data_set_name
```

where *data_set_name* is the name of the data set containing the processor operations control file built by the customization dialog. For details on the ISQSTART command, refer to *System Automation for z/OS Operator's Commands*.

2. You stop processor operations by issuing **ISQSTOP** from the NetView operator console.

Starting and Stopping Applications

Figure 7 on page 59 shows the startup processing of an application under the control of SA z/OS.

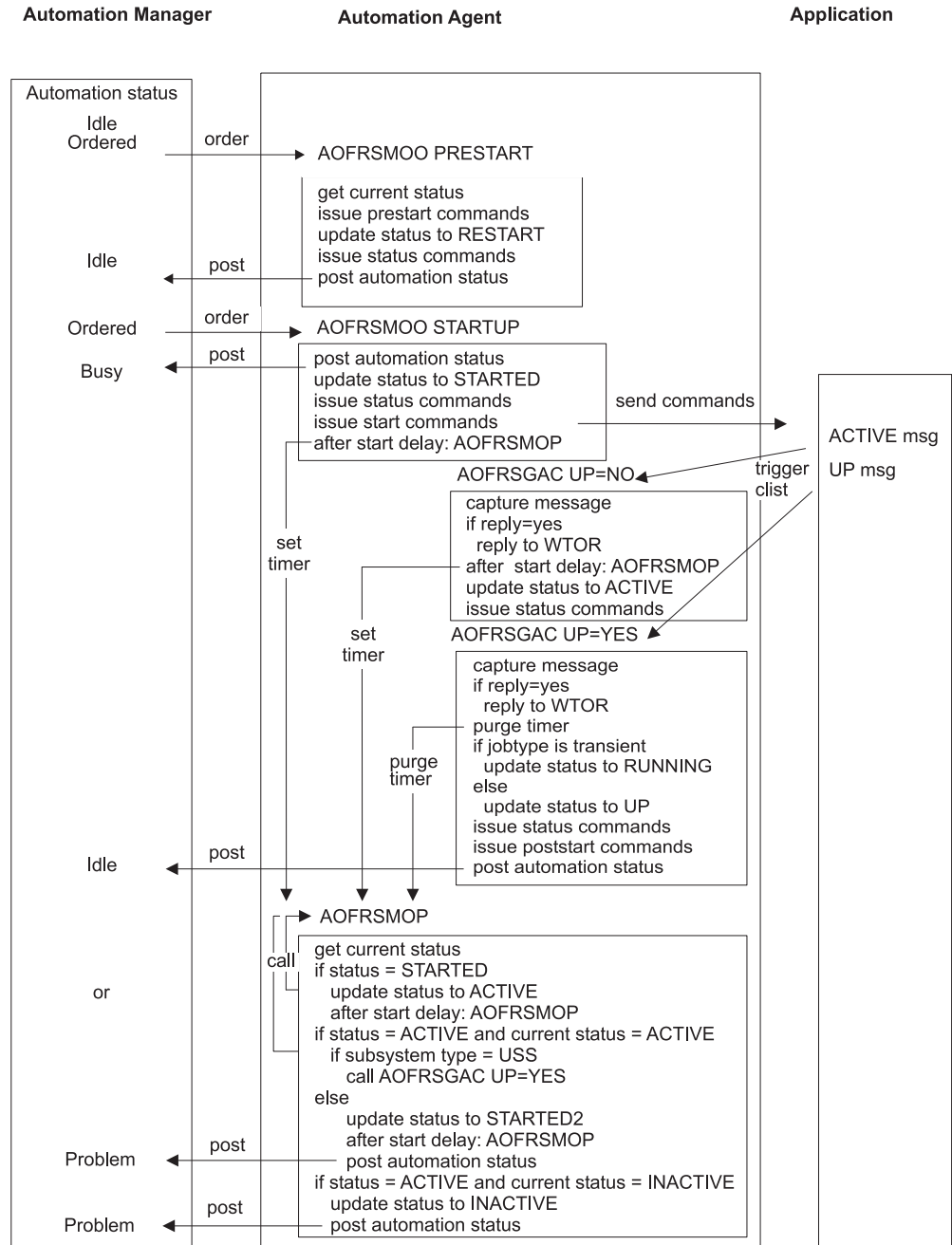


Figure 7. Application Startup Under SA z/OS's Control

Actions such as issuing prestart, shutdown, or status commands are only processed if the appropriate automation flag is ON.

The REXX procedure AOFRSGAC is also known under its synonym ACTIVMSG.

Figure 8 on page 60 shows the stop processing of an application under the control of SA z/OS.

Starting and Stopping Applications

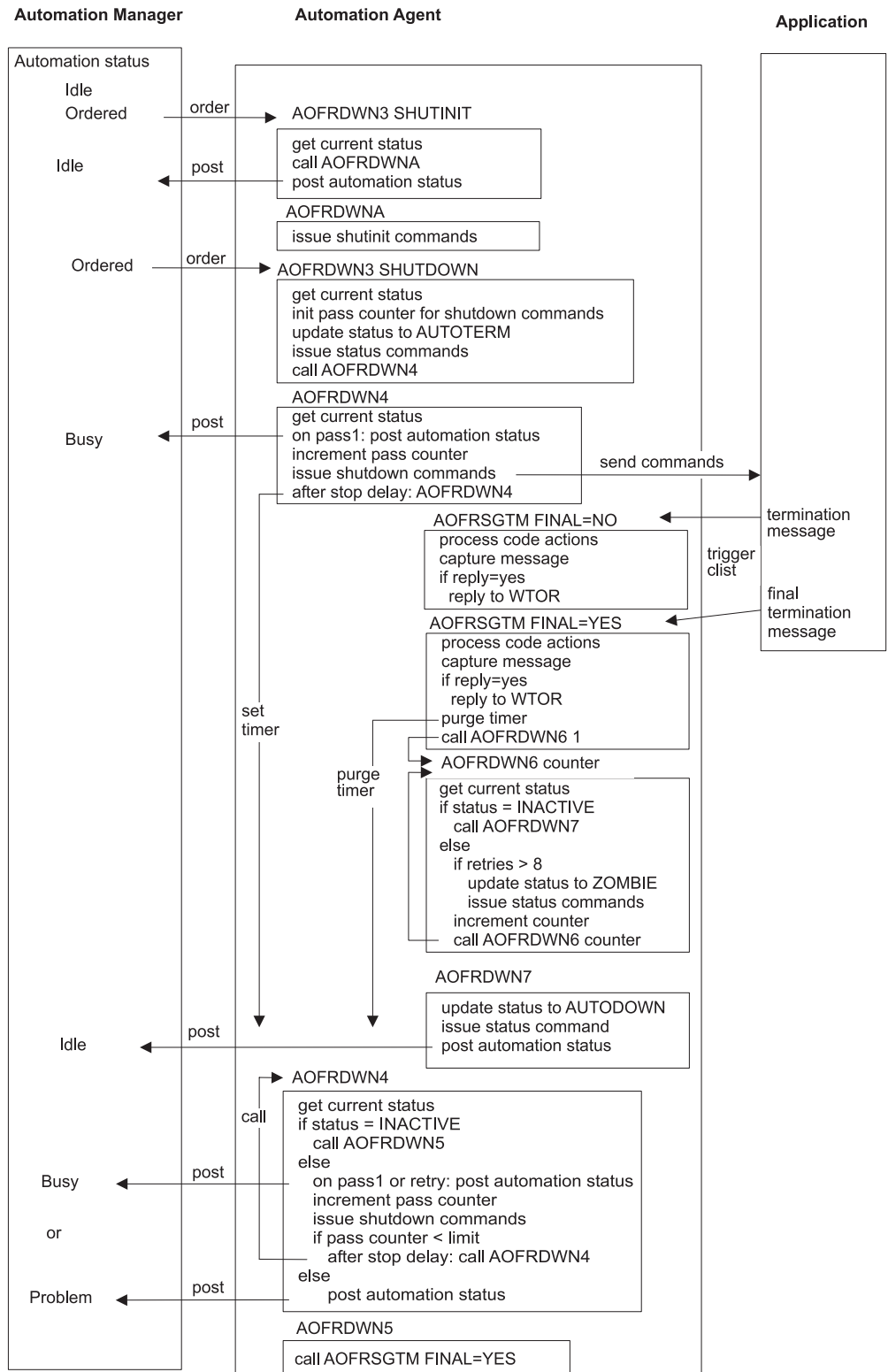


Figure 8. Application Stop Under SA z/OS's Control

Actions such as shutinit, shutdown, or status commands are only processed if the appropriate automation flag is ON.

The REXX procedure AOFRSGTM is also known under its synonym TERMMSG.

Chapter 4. Refreshing Automation Policy

The primary method to refresh the automation policy is to issue the command:

```
INGAMS REFRESH
```

This command operates sysplex-wide. All systems in the sysplex are refreshed automatically. It supports incremental updates on a system basis, that is, only the data that is affected by changes in the policy database is refreshed.

Besides the method of using the INGAMS command with the REFRESH option, there are other ways to perform the task of refreshing the automation policy. These are described in this chapter.

To help you dynamically change policy after you have initialized SA z/OS on your system, SA z/OS lets you

- load a different automation control file
- reload an automation control file that you have changed
- restore an automation control file that has been previously saved into a warm start cache.

The automation control file load is normally performed at system operations initialization (NetView start). It is automatically triggered, if the automation manager configuration file (AMC file) is changed. It can also be triggered from an operator by entering an ACF LOAD/REFRESH command. You do normally not have to know the correct automation control file name. It is automatically determined by SA z/OS.

At an ACF load the automation manager is contacted and queried for the 'Configuration Token' to make sure that the ACF that will be loaded is compatible with the AMC file.

If you issue an **ACF** command and there is no automation control file that is compatible with the AMC file, the load is not performed and an error message is issued.

The ACF REFRESH is able to do incremental updates of an ACF that is either in storage or in the save/restore cache. An ACF load will normally not be necessary, in most cases an ACF REFRESH is sufficient. The initialization processes the different data on several autotasks in parallel. Therefore there is no need to disable automation during a refresh.

Note: For a description of the complete ACF syntax, refer to *System Automation for z/OS Operator's Commands*.

To reload or replace the current NetView automation table after making changes to it, type the following on any NetView command line:

```
autotbl member=tablename
```

where *tablename* is the name of the current NetView automation table.

Note: If the NetView automation table you nominate does not contain the entries that SA z/OS needs, you will get unpredictable results.

Refreshing an Automation Manager Configuration

See *Tivoli NetView for z/OS Command Reference* for more information on the AUTOTBL command.

Refreshing an Automation Manager Configuration

The REFRESH subcommand of the MVS MODIFY (F) command may be used to initiate a configuration refresh operation, and is effective only when directed to the primary automation manager.

```
F jobname  
,REFRESH,new_data_set_name
```

The *new_data_set_name* may be specified in several forms as follows:

- *dsname* - This form specifies the fully qualified data set name to be used.
- *gdgname(generation)* - This form specifies a generation data group name and the generation group member to be used. The generation number is relative to the most recently created member, and may be 0 (for the most recent) or a negative number (-1,-2, ..., -n) where n is the number of generations specified when defining the GDG minus 1.
- * (Asterisk) - This specifies that the refresh operation should use the same data set (and GDG member) as was used on the last successful refresh operation or during automation manager initialization.
- *(generation) - This specification form specifies that a member of the same GDG used for the previous refresh operation is to be used. The generation number is relative to the most recently created GDG member at the time of command entry.

Use this command when no automation agent is running.

Example:

```
REFRESH,*(0) - use most recent generation of the specified GDG
```

Identifying the Current Automation Control File and NetView Automation Table

The automation control file and the NetView automation table are the two files that control system-level automation. The automation control file contains SA z/OS automation policy, which specifies what, when, and how to automate. The NetView automation table specifies which automation procedure to run when various messages are received.

Some sites use different automation control files and NetView automation tables for different occasions. For example, a site may use one NetView automation table during normal operations and another during holiday weeks. If your site has more than one automation table, you may need to determine which one is active.

An automation control file and NetView automation table are made available during SA z/OS initialization. The NetView automation table is loaded into NetView, and the automation control file can be either loaded into SA z/OS or restored from a warm start cache.

To identify the currently active automation control file (ACF) and NetView automation table you can use the command dialog DISPSYS. See Figure 9 on page 63 for an example.

Identifying the Current AMC File Used by the Automation Manager

```
AOFKADAE          SA z/OS - Command Dialogs      Line 1    of 42
Domain ID = IPSFM  ----- DISPSYS -----      Date = 04/12/04
Operator ID = OPER1      System = KEY1          Time = 14:25:16

System           : KEY1          in Sysplex   : KEY1PLEX
Domain           : IPSFM
Sysplex Group    : SYSPLEX_GROUP_01
XCF Group name   : INGXSGOY

Software
  Operating System : z/OS 01.05.00
  NetView          : Tivoli NetView for z/OS V5
  Tower(s)        : SA MVSCMDMGT
  System Automation : V2R3M0

Configuration
  Data set        : OPER1.USER.KEYPLEX.ACF(ACFZ995)
  Built by        : OPER1 04/11/04 10:20:36
  Activated       : 04/11/04 10:21:19
  CFG Token       : 200404051055068A0345169672

Message automation table(s)
DSITBL01 INGMMSG01

Flags
  Automation      : No
  Init Start      : No
  Start           : No
  Recovery        : No
  Shut Down       : No
  Restart         : No

Scheduling Subsystem : JES2
Type                 : JES2
Root for SDF updates : KEY1
SDF focal point      : IPSFM
  primary focal point : IPSFM
  backup focal point  : IPSFM

Sysplex Timing
  Defined         : No timers
  Current         : Unknown
  ID's           : None

Heartbeat Interval   : 5 Minute(s)
Missing Heartbeat Delay : 30 Seconds
```

Figure 9. Command Dialog DISPSYS

Identifying the Current AMC File Used by the Automation Manager

To identify the current automation manager configuration file, you can use the diagnostic facilities of the INGAMS command (see also “How to Use the Diagnostic Option of INGAMS” on page 277). From the INGAMS command dialog, you can select option *B Show Details* for an automation manager. A panel similar to the following is displayed, where relevant information concerning the AMC file is shown.

Identifying the Current AMC File Used by the Automation Manager

```
INGKYAM3                SA z/OS - Command Dialogs      Line 1    of 24
Domain ID = IPSFM      ----- INGAMS -----      Date = 04/12/04
Operator ID = OPER1    Sysplex = KEY1PLEX          Time = 15:11:16

Automation Manager : KEY2$$$1      running on system : KEY2
Operation mode     : PAM           Job name           : AMOY
Status            : READY
XCF-Groupname     : INGXSGOY
Start time        : 12 APR 2004 12:50:27
PAM selected time : 12 APR 2004 12:50:32

Takeover file     : OPER1.USER.HSATKOV

Logic Deck ...
Release           : V2R3M0
Date built        : 26 May 2004
Time built        : 11:48:04
Last APAR         : NONE

Config dataset name : OPER1.USER.KEYPLEX.ACF
Config member      : HSAZ999
                        Z999CLGC  STRUCTURE  20040405105506
                        Z999CRES  STRUCTURE  20040405105506
                        Z999CSCH  SCHEDULE   20040405105506

Config token       : 200404051055068A0345169672
Config version    : 01

Diagnostic Info
Snapshot size     : 3492180
Number of resources : 277
Number of requests : 65
Number group requests : 62
```

Figure 10. Command Dialog INGAMS

Part 2. Using SA z/OS on the Host

Chapter 5. Getting Started

Different sets of commands are available from host sessions:

- automation manager commands
- system operations commands (automation agent commands)
- processor operations commands
- I/O operations commands

With system operations commands, you can control and maintain all of the resources sysplex-wide from a single point of control. They operate in two modes:

- Fullscreen mode: If it is a command to display information and you just enter the command name, a panel is displayed showing all available resources. On this panel, you can specify further actions/commands for a special resource. If you enter a command to maintain a resource without further parameters, a fullscreen panel prompts you for more information.
- Line mode: From a command line, you can enter the complete syntax of a command to receive either the desired output directly or to manipulate the resource you wanted in the way you wanted.

All system operations commands (automation agent commands) also operate in fullscreen mode and line mode unless stated in the reference documentation that the OUTMODE parameter is not supported.

The tasks that you can manage with the available commands are described in more detail in the subsequent chapters and sections. If you need the complete syntax of these commands, refer to *System Automation for z/OS Operator's Commands*.

Issuing Commands

With SA z/OS, there are several ways that you can issue a command:

- You can use command dialogs to enter command parameters and issue commands using 3270 panels. See "Using Command Dialogs" on page 68.
- You can use an object's popup menu on the NMC workstation.

SA z/OS adds a selection of predefined processor operations and system operations commands to the NetView Management Console. This enables you to use the NetView Management Console as a front end for managing your z/OS host environment.

The sets of available commands are provided in corresponding response files. If you want to add or remove commands from the delivered sets, you can use the *Command Profile Editor (CPE)*. For information on how to use this editor, refer to *NetView Management Console User's Guide*.

For information about how to use this graphical command interface, refer to Chapter 11, "Using the NetView Management Console for SA z/OS," on page 217.

- You can enter I/O operations commands at the z/OS operator console. See "Using the z/OS Operator Console for I/O Operations Commands" on page 204.

Issuing Commands

- You can issue processor operations, system operations, and automation manager commands from the same interface via NetView terminals as described in “Using Command Dialogs.”
- I/O operations commands can be issued from ISPF dialog panels as described in “Using the I/O Operations ISPF Dialog” on page 204.

For each method, instructions on the window or panel tell you what to do. The commands are described in greater detail in *System Automation for z/OS Operator’s Commands*.

Using Command Dialogs

To start SA z/OS, type A0C from the NetView command line in your NetView session to reach the *System Operations MAIN MENU* shown in Figure 11.

```
AOFPOPER          SA z/OS - System Operations
Domain ID  = IPUFM  ----- MAIN MENU -----   Date = 06/13/04
Operator ID = OPER1                               Time = 13:17:14

      Select          Description                Component

      1              Operator Interface          OPER
      2              Command Dialogs            CD
      3              Status Display Facility    SDF
      I              IMS Automation Feature     IMS
      C              CICS Automation Feature   CICS
      O              OPC Automation Feature    OPC
      L              User defined Local Functions

SA z/OS Version 2 Release 3
Licensed Materials - Property of IBM
5645-006 (C) Copyright IBM Corp. 1990, 2004 All Rights Reserved.

Command ==>
PF1=Help      PF2=End      PF3=Return      PF6=Ro11
               PF12=Retrieve
```

Figure 11. System Operations Main Menu

Type 2 or CD on the command line to invoke the *Command Dialogs Main Menu* (Figure 12 on page 69). Starting with this menu, you can interact with 3270-type panels to issue SA z/OS system operations commands.

You can enter a command with two methods:

- Type the number or letter corresponding to the desired command on the command line,
- or type the name of the command. If you choose to type a command, you can add options to it.

For example, if you type number 2 into the command line of the *Command Dialogs Main Menu*, you reach the AOCTRACE command dialog which prompts you for further options.


```

AOFPOCIM          SA z/OS - Command Dialogs          Line 1 of 26
Domain ID   = IPUFM   ----- Main Menu -----      Date = 06/13/04
Operator ID = OPER1                                     Time = 13:49:03

   1  AOCHHELP      Displays or executes System Operations commands
   2  AOCTRACE      Turn Debugging feature on or off
   3  DISPAUTO      Display automation statistics
   4  DISPSTAT      Display current automation status of subsystem(s)
   5  DRAINJES      Initiate DRAIN process for JES2 controlled devices
   6  INGAMS        Operate the automation manager (stop/modify op mode)
   7  INGAUTO       Turn automation on/off for resource(s)
   8  INGCF         Control coupling facility resources
   9  INGGROUP      Manipulate Groups
  10  INGHC         Show Health Checker results
  11  INGHIST       Show history information about workitems
  12  INGLIST       Display list of resources
  13  INGLKUP       Display unknown jobs and STCs
  14  INGNTFY       Change status of operator notifications
  15  INGPLEX       Display system information of all systems in the sysple
  16  INGRPT        Display Statistics
  17  INGRELS       Show relationships and evaluation for a resource
  18  INGREQ        Initiate shutdown or startup process for a resource
  19  INGSCHED      Display all defined schedules (service periods)
  20  INGTHRES      Displays/sets thresholds for resource(s)
  21  INGTRIG       Display all triggers within automation manager scope
  22  INGVOTE       Display all requests and pending votes
  23  SETASST       Turn assist on/off for resource(s)
  24  SETSTATE      Change state of subsystem(s)
  25  SETTIMER      Displays current timer(s)
   X  EXIT          Leave SA for z/OS Command Dialogs

Command ==>
PF1=Help      PF2=End      PF3=Return      PF6=Ro11
               PF8=Forward      PF12=Retrieve

```

Figure 12. SA z/OS Command Dialogs Main Menu

Some of the commands operate sysplex-wide; especially those sent to the automation manager. For other commands you can specify a target system on which they should operate, for example, if you want to display the automation flags on a certain target system.

The commands present a snapshot of the data, taken at the time the command is executed. Some data may be refreshed when other commands are issued from the same panel. To force a refresh of all data you must press PF9.

How to Get Help

In case you need help, you have the following possibilities:

- Get dialog help: On a NetView console, type AOCHHELP or ISQHELP to see the main help panel.
- On an SA z/OS panel, press PF1 for a description of the panel and its fields.
- Get help for a message: On a NetView console, type HELP or HM and the message identifier on a NetView console. For example, type :
HELP A0F001
- Get information about automation agent and automation manager status: Type EXPLAIN *status_name*, for example EXPLAIN UP. Or type EXPLAIN *resource_name* to get information about the automation agent status of the specified resource. Type help *amstatus* to see information about the automation manager status.
- Get command help: On the NetView console, type
HELP commandname

Using Command Dialogs

Chapter 6. How to Get Resource Information Sysplex-Wide

This chapter describes commands that you can use to retrieve information from all resources defined within a sysplex from a single point of control. The commands are presented here in an overview. Each command is documented with sample screens and the complete syntax in *System Automation for z/OS Operator's Commands*.

SA z/OS provides commands that let you look at resources from two different point of views:

- from the point of view as seen by the automation manager; this lets you see which state a resource is in or which votes are currently active against this resource. These commands are described in "Commands that Are Processed by the Automation Manager"
- from the point of view as seen by the automation agents; this lets you see thresholds or startup and shutdown commands defined for the resources. These commands are described in "Commands that Are Processed by the Automation Agents" on page 76.

Commands that Are Processed by the Automation Manager

The automation manager decides how automation is performed for the known resources in the sysplex. To decide how to handle the requests that it receives through the various channels (for example, the defined automation policy or interactive operator requests), the automation manager uses an internal hierarchy of priorities which are listed in Table 3 on page 15.

SA z/OS offers a series of commands with which you can find out which information the automation manager currently holds about resources, for example, relationship information or information about the status of resources and whether schedules exist for them. These commands are introduced in the following sections.

How to Use INGLIST to View Resources

This command provides the details about the resources in the sysplex from the point of view of the automation manager. The following information is displayed for a resource (see Figure 13 on page 72):

- Name and type of the resource
- Compound status
- Current status (observed)
- Desired status
- Automation status
- Startability status
- Health status
- Automation flag
- Hold flag
- Description
- Schedule name, if any
- Trigger name, if any

Commands that Are Processed by the Automation Manager

- Start type
- Stop type
- Resource category and nature

```

INGKYST0          SA z/OS - Command Dialogs          Line 1 of 284
Domain ID = IPUFM          ----- INGLIST -----          Date = 03/10/04
Operator ID = OPER1          System = AOC7          Time = 16:18:15
CMD: A Update      B Start      C Stop      D INGRELS  E INGVOTE  F INGINFO
   G Members      H DISPTRG  I INGSCHED J INGGROUP M DISPMTR / scroll
CMD Name          Type System    Compound   Desired   Observed   Nature
-----
AGFG              APL AOC7     INHIBITED AVAILABLE SOFTDOWN
- AGFG2           APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- AOC7            SYS AOC7     PROBLEM    AVAILABLE PROBLEM    BASIC
- AOC7            SYS AOC7     SATISFACTORY AVAILABLE AVAILABLE
- APPC            APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- ASCH            APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- BATB000P100     APL AOC7     PROBLEM    AVAILABLE HARDDOWN
- BATB000P200     APL AOC7     PROBLEM    AVAILABLE HARDDOWN
- BATB000P300     APL AOC7     PROBLEM    AVAILABLE HARDDOWN
- BATB000S100     APL AOC7     PROBLEM    AVAILABLE HARDDOWN
- BATB000S200     APL AOC7     PROBLEM    AVAILABLE HARDDOWN
- BATB000S300     APL AOC7     PROBLEM    AVAILABLE HARDDOWN
- BLSJPRMI        APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- BNK#$$X$X       APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- BNK#$$X$X       APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- BNKBAGT          APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- BNKCICSPA        APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- BNKDB            APL AOC7     SATISFACTORY AVAILABLE AVAILABLE
- BNKDEMO1         APL AOC7     PROBLEM    AVAILABLE HARDDOWN
- BNKDEMO2         APL AOC7     PROBLEM    AVAILABLE HARDDOWN
- BNKDEMO3         APL AOC7     PROBLEM    AVAILABLE HARDDOWN

Command ==>
PF1=Help   PF2=End   PF3=Return  PF4=DISPSTAT  PF5=Filters  PF6=Ro11
            PF8=Forward  PF9=Refresh PF10=Previous PF11=Next    PF12=Retrieve
  
```

Figure 13. INGLIST Command Dialog

You can see more information about the resources by scrolling horizontally using PF11. Some of this information, especially the automation status, changes in real time. By pressing PF9 (Refresh), you can obtain a new set of data for the displayed subsystems.

On this panel, you can use PF5 to invoke a filter dialog (which is equivalent to using the INGFLT command available from any command line). With this filter, you can reduce the amount of information shown in the display. It enables you to setup a specific operator view for an operator. For example, you can set up a filter that shows only certain resources in specified observed and desired states.

You can also use the INGLIST command dialog to launch other actions that are related to resources or groups, that means, you can also this command to maintain and control resources.

The CMD column on the left hand side of the screen allows you to issue any of the commands listed across the top of the screen against any of the resources that are displayed. Enter the letter corresponding to the command beside the resource you want it issued against and press enter. When you return from the command there may be a slight pause as the display refreshes itself.

The available commands are indicated at the top of the INGLIST panel:

```

A Update      B Start      C Stop      D INGRELS  E INGVOTE  F INGINFO
G Members     H DISPTRG   I INGSCHED J INGGROUP M DISPMTR / scroll
  
```

Commands that Are Processed by the Automation Manager

There is now a new command code, M, that shows monitors that are associated with resource (that is, it invokes the DISPMTR command).

Note: If you have defined exit routine AOFEXC04, there is an additional action code U. See *System Automation for z/OS Customizing and Programming* for details.

On the INGLIST panel, you can use PF4 to toggle to the DISPSTAT command dialog, which gives you the automation agent view of the resources in the sysplex (see “How to Display the Automation Agent View with DISPSTAT” on page 77). To return to the INGLIST panel press the PF3 key on the DISPSTAT panel. Note that pressing PF4 on the DISPSTAT panel displays the INGLIST panel for all resources shown in the DISPSTAT panel but this might not be the same as the original INGLIST display since the DISPSTAT display might have been modified due to filtering or suppressing of resources that the automation agent does not handle.

You can use PF5 to define a filter as described in “How to Specify Filters.”

How to Specify Filters

Figure 14 shows the INGFLT command dialog where you can specify or revise filter criteria.

```
INGKYFLT          SA z/OS - Command Dialogs
Domain ID   = IPUFM   ----- INGFLT -----   Date = 05/06/04
Operator ID = OPER1                               Time = 13:22:14

Specify or revise the filter criteria:

Resources...      format: name/type/system or *am*/*/*
  AGFG*/*/*
-----

Observed status ==> AV
Desired status  ==> *
Automation status ==> *
Compound status ==> *
Health status   ==> *
Automation flag ==> _____ Yes or No
Resource category ==> *
Resource subtype ==> *
Group type      ==> *

Command ==>
PF1=Help  PF2=End  PF3=Return  PF4=Clear  PF5=Reset  PF6=Ro11
          PF9=Save                PF12=Retrieve
```

Figure 14. INGFLT Command Dialog

The panel from Figure 14 shows the current INGLIST filters. The filter settings are saved, so when you next run INGLIST they will be used again. Press PF9 to save your currently displayed filter criteria.

You can specify one or more of the following filters:

Resources

This is the list of resources in name/type/system format. You can also specify a wildcard.

Commands that Are Processed by the Automation Manager

Observed status

This is the list of observed statuses. You can abbreviate the status, for example "av" for Available. Any resource whose observed status is in the list is eligible for display.

Desired status

This is the list of desired statuses. You can abbreviate the status. Any resource whose desired status is in the list is eligible for display.

Automation status

This is the list of automation statuses. You can abbreviate the status. Any resource whose automation status is in the list is eligible for display.

Compound status

This is the list of compound statuses. You can abbreviate the status. Any resource whose compound status is in the list is eligible for display.

Health status

This is the list of health statuses. You can abbreviate the status. Any resource whose health status is in the list is eligible for display.

Automation flag

This is the automation flag for the resource. Specify YES or NO.

Resource category

This is the list of categories the resource belongs to. This is one of the following: CICS, IMS, DB2, OPC, JES2, JES3, USS, or IMAGE.

Resource subtype

This specifies the region subtype for CICS, DB2 and IMS or the controller type for OPC applications, for example, AOR for *CICS* or IRLM for *DB2*.

In order to be eligible for the display, a resource must match all filter criteria. An asterisk means that the filter is not set.

Use the PF4 key to clear the currently established filter settings. The filter will then contain an asterisk, meaning that the filter is not set, or a blank. Use the PF5 key to go back to the currently established filter settings.

Note: If INGFLT was called from INGLIST, then the filters that are displayed will take effect when you press ENTER. If you press PF9 first, then the filters will be saved.

How to Use INGINFO to View a Selected Resource

The INGINFO command displays details about a specified application or application group as seen by the automation manager. If you enter this command without parameters, a dialog prompts you for more information. For a detailed description of the command syntax refer to *System Automation for z/OS Operator's Commands*.

You can also issue this command for a certain resource displayed in the result panel of the INGLIST command. In the resulting panel (an example is shown in Figure 15 on page 75), you will see information about the resource's different statuses, its automation flag settings, whether dependencies are satisfied and which votes are currently active against the resource.

Commands that Are Processed by the Automation Manager

```

INGKYING          SA z/OS - Command Dialogs          Line 1 of 220
Domain ID = IPUFM ----- INGINFO -----          Date = 05/13/04
Operator ID = OPER1          Sysplex = AOC7PLEX          Time = 12:53:10

Resource ==> APPC/APL/AOC7          format: name/type/system
System ==>          System name, domain ID or sysplex name

Resource      : APPC/APL/AOC7
Description   : APPC/MVS automation policy.

Status...    :
Observed Status : AVAILABLE
Desired Status  : AVAILABLE
Automation Status : IDLE
Startable Status : YES
Compound Status  : SATISFACTORY
Health Status   : NORMAL

Dependencies  :
PreStart      : Satisfied
Start         : Satisfied
PreStop       : Satisfied
Stop          : Unsatisfied
Startability  : Satisfied

Current Order : -None-

Flags...     :
Automation    : YES
Hold          : NO

Commands...  :
Start type    : -None-
Stop type     : -None-

Trigger      : -None-
Schedule     : -None-

Backward Relationships :
AOC7/SYG/AOC7          HasMember
ASCH/APL/AOC7          HasParent

Forward Relationships  :
VTAM/APL/AOC7          HasParent

Requests      :
Votes         :

Origin...     :
Nature        : Resource
Owner         : SA z/OS Resources
Loaded from   : SATEST.WORK.ACF(Z995CRES)

History      :
2004-05-13 04:58:26
Workitem ID = 07D0070D023A69460001
HSAL6104I Linked to trigger: HsaNMCobserver
2004-05-13 04:54:58
Workitem ID = 07D0070D0236E53D0001
HSAL6105I Unlinked from trigger: HsaNMCobserver

Command ==>
PF1=Help      PF2=End      PF3=Return      PF6=Ro11
               PF8=Forward  PF9=Refresh    PF12=Retrieve
    
```

Figure 15. INGINFO Command Dialog

On the INGINFO panel, you can use PF4 to toggle to the DISPINFO command dialog, which gives you the automation agent point of view of the current resource (see “How to Use DISPINFO to Display Detailed Information” on page 81).

How to Display Relationship Information: INGRELS

The INGRELS command displays a panel that shows the relationships that are defined for the specified resource and the status of the current conditions (whether they are satisfied or unsatisfied).

```

INGKYRL0          SA z/OS - Command Dialogs      Line 1   of 13
Domain ID  = IPUFM  ----- INGRELS -----      Date = 03/15/04
Operator ID = OPER1      System = AOC7           Time = 10:11:42

Resource ==> APPC/APL/AOC7
System    ==> _____ System name, domain id or sysplex name

Cmd:  - Collapse  + Expand  E INGVOTE  F INGINFO  J INGGROUP  S Focus

Cmd Name      Type System  Dir Relationship
-----
   AOC7        SYG  AOC7    B  HasMember
              Cond: startDependency
              Cond: stopDependency
              Cond: extStart - UNSATISFIED
              Cond: extStop - UNSATISFIED
              Cond: extStopDelayed
              Cond: prestartDependency
              Cond: prestopDependency

   ASCH        APL  AOC7    B  HasParent - Strong
              Cond: stopDependency - UNSATISFIED

   VTAM        APL  AOC7    F  HasParent
              Cond: startDependency
              Cond: startable

Command ==>
PF1=Help      PF2=End          PF3=Return      PF5=Collapse all  PF6=Roll
               PF9=Refresh      PF11=Expand all  PF12=Retrieve
    
```

Figure 16. INGRELS Command Dialog

The INGRELS command dialog shows the relationships that have been defined for the current resource in both directions (in the example from Figure 16 for resource *APPC/APL/AOC7*). So in column *Dir* from the INGRELS command dialog shown in Figure 16, there is:

- A *B* for backward relationships; that means the relationship exists from the partner resource to the specified resource
- An *F* for forward relationships; that means the relationship exists from the specified resource to the partner resource

Relationship conditions are also shown (with signature *Cond:*) and whether they are unsatisfied.

The level of detail that is displayed can be changed using the following keys:

- + Shows the condition details that exist for the selected relationship. This is only applicable if the relationship conditions were previously collapsed.
- Collapse the condition details. Only the relationship will be shown.

Commands that Are Processed by the Automation Agents

An automation agent is installed on each system upon which resources reside that need to be automated by SA z/OS. For example, automation agents are responsible for having the appropriate commands processed for subsystem startup and shutdown, reporting status updates to the automation manager and message automation. Automation agents access information for their tasks from the automation control file.

Commands that are Processed by the Automation Agents

SA z/OS offers some commands with which you can acquire information about resources running on the system where the automation agent resides.

How to Display the Automation Agent View with DISPSTAT

The DISPSTAT command displays fullscreen panels showing the following information for resources defined to the current automation agent:

- subsystem name
- automation status (not dynamically refreshed)
- the system the instance of the subsystem is running on
- job name
- current automation flag settings
- the type (MVS, Transient, One Shot, or Non-MVS) of the subsystem
- which trigger, if any, is associated with the subsystem; this information can only be seen for down level SA z/OS systems.
- which service period, if any, is associated with the subsystem; this information can only be seen for down level SA z/OS systems.
- the date and time its automation status last changed
- the date and time its automation status was last confirmed by active monitoring
- a user supplied description of the resource

Status conditions are color-coded to make it easier to identify problems.

If you have run DISPSTAT without filtering and you did not provide any input parameters, a panel similar to the following is displayed:

```
AOFKSTA5          SA z/OS - Command Dialogs          Line 1 of 441
Domain ID = IPUFM  ----- DISPSTAT -----          Date = 03/30/04
Operator ID = OPER1                               Time = 17:25:32
A ingauto B setstate C ingreq-stop D thresholds E explain F info G tree
H trigger I service J all children K children L all parents M parents
CMD RESOURCE      STATUS   SYSTEM  JOB NAME  A I S R D RS TYPE  Activity
-----
- HUGSGX           UP       AOCA    HUGSGX   Y Y Y Y Y Y MVS   --none--
- HUGSHUTINIT     UP       AOCA    HUGSHUTI Y Y Y Y Y Y MVS   --none--
- HUGSRVGRPT2     ENDED   AOCA    HUGSGT2  Y Y Y Y Y Y ONE SHOT --none--
- HUGSRVGTRA1     DOWN    AOCA    HUGSGTR1 Y Y Y Y Y Y MVS   --none--
- HUGSTART        INACTIVE AOCA    HUGSTART Y Y Y Y Y Y MVS   --none--
- HUGTRIG1        DOWN    AOCA    HUGTRIG1 Y Y Y Y Y Y MVS   --none--
- HUGWTSO         UP       AOCA    HUGWTSO  Y Y Y Y Y Y MVS   --none--
- HUGWVTAM        UP       AOCA    HUGWVTAM Y Y Y Y Y Y MVS   --none--
- IRRDPTAB        ENDED   AOCA    IRRDPTAB Y Y Y Y Y Y ONE SHOT --none--
- JESWTR          ENDED   AOCA    JESWTR   Y Y Y Y Y Y ONE SHOT --none--
- JES2            UP       AOCA    JES2     Y Y Y Y Y Y MVS   --none--
- JLP111A         UP       AOCA    JLP111A  Y Y Y Y Y Y MVS   --none--
- JLP111B         UP       AOCA    JLP111B  Y Y Y Y Y Y MVS   --none--

Command ==>
PF1=Help   PF2=End   PF3=Return PF4=INGLIST PF5=Filters PF6=Ro11
PF7=Back   PF8=Forward PF9=Refresh PF10=Previous PF11=Next PF12=Retrieve
```

Figure 17. DISPSTAT Command Dialog

Note: If you have defined exit routine AOFEXC04, there is an additional action code U. See *System Automation for z/OS Customizing and Programming* for details.

The panel from Figure 17 does not show all of the data that is available. To access the next panel, press PF11.

Commands that are Processed by the Automation Agents

```

AOFKSTA5          SA z/OS - Command Dialogs          Line 1 of 441
Domain ID = IPUFM ----- DISPSTAT -----         Date = 03/30/04
Operator ID = OPER1                                     Time = 17:26:25
A ingauto B setstate C ingreq-stop D thresholds E explain F info G tree
H trigger I service J all children K children L all parents M parents
CMD RESOURCE      STATUS      LAST CHANGED      LAST MONITORED
-----
- HUGSGX           UP           03/29/04 20:41:38 03/30/04 17:10:49
- HUGSHUTINIT     UP           03/29/04 20:40:59 03/30/04 17:10:49
- HUGSRVGRPT2     ENDED       03/29/04 20:29:45 03/30/04 17:10:49
- HUGSRVGTRA1     DOWN        03/29/04 20:29:45 03/30/04 17:10:49
- HUGSTART        INACTIVE    03/30/04 14:46:51 03/30/04 17:10:49
- HUGTRIG1        DOWN        03/29/04 20:29:47 03/30/04 17:10:49
- HUGWTSO         UP           03/30/04 14:36:48 03/30/04 17:10:49
- HUGWVTAM        UP           03/30/04 14:36:49 03/30/04 17:10:49
- IRRDPTAB        ENDED       03/29/04 20:29:50 03/30/04 17:10:49
- JESWTR          ENDED       03/29/04 20:29:50 03/30/04 17:10:49
- JES2            UP           03/29/04 20:29:51 03/30/04 17:10:49
- JLP111A         UP           03/29/04 20:52:17 03/30/04 17:10:49
- JLP111B         UP           03/29/04 20:49:41 03/30/04 17:10:49
-
Command ==>
PF1=Help  PF2=End    PF3=Return  PF4=INGLIST  PF5=Filters  PF6=Roll
PF7=Back  PF8=Forward PF9=Refresh  PF10=Previous PF11=Next   PF12=Retrieve

```

In cases where a wide (132 column) display is used, all of the data can be displayed on two screens. On a smaller 80 column display there is a third screen, which displays the Trigger, Service Period and Description data.

Trigger and Service Period information is only shown when SA OS/390 1.3 resources are displayed.

Use PF11 to cycle through this as well to see the following information.

```

AOFKSTA5          SA z/OS - Command Dialogs          Line 1 of 441
Domain ID = IPUFM ----- DISPSTAT -----         Date = 03/30/04
Operator ID = OPER1                                     Time = 17:26:51
A ingauto B setstate C ingreq-stop D thresholds E explain F info G tree
H trigger I service J all children K children L all parents M parents
CMD RESOURCE      STATUS      TRIGGER      SCHEDULE DESCRIPTION
-----
- HUGSGX           UP           -            -            only test for RR
- HUGSHUTINIT     UP           -            -            Application with shutinit command
- HUGSRVGRPT2     ENDED       -            -            Server group with transients
- HUGSRVGTRA1     DOWN        -            -            Server group with transients
- HUGSTART        INACTIVE    -            -            HUG application with prestart...
- HUGTRIG1        DOWN        -            -            HUG application with triggers
- HUGWTSO         UP           -            -            resource for combination move-bas
- HUGWVTAM        UP           -            -            resource for combination move-bas
- IRRDPTAB        ENDED       -            -            RACF dynamic parsing table init a
- JESWTR          ENDED       -            -            Write SYSLOG after IPL
- JES2            UP           -            -            JES2 application for each test sy
- JLP111A         UP           -            -            MAWA/P
- JLP111B         UP           -            -            MAWA/P
-
Command ==>
PF1=Help  PF2=End    PF3=Return  PF4=INGLIST  PF5=Filters  PF6=Roll
PF7=Back  PF8=Forward PF9=Refresh  PF10=Previous PF11=Next   PF12=Retrieve

```

Some of this information, especially the automation status, changes in real time. By pressing PF9 (Refresh), you can obtain a new set of data for the displayed subsystems.

You can use PF4 to toggle to the INGLIST panel to see the same resources as shown in the DISPSTAT panel but now from the automation manager point of

Commands that are Processed by the Automation Agents

view. To return to the DISPSTAT panel press the PF3 key on the INGLIST panel. Note, pressing PF4 on the INGLIST panel displays the DISPSTAT panel for all resources shown in the INGLIST panel but this might not be the same as the original DISPSTAT display since the INGLIST display might have been modified due to filtering.

From the DISPSTAT display you can initiate a number of other commands:

INGAUTO

This lets you change the subsystems automation flags

SETSTATE

This lets you change the subsystems automation status if it is inactive.

INGREQ - STOP

This lets you shut down the subsystem.

thresholds - INGTHRES

This lets you review the subsystems threshold settings.

EXPLAIN

This presents an explanation of the subsystems current automation status.

DISPINFO

This presents a detailed display of the subsystem.

DISPTREE

This presents a view of the target systems parent/child graph, focused on the subsystem.

DISPTRG

This shows details of whichever trigger, if any, is associated with the subsystem.

INGSCHEM

This shows details of the service period that has been defined for the subsystem.

All of these commands work in the same way as DISPSTAT - fetching it from the target system and presenting it to you on a full screen panel.

On the DISPSTAT panel, you can use PF5 to invoke a filter dialog which is equivalent to using the DISPSFLT command available from any command line (see Figure 18 on page 80). With this filter, you can reduce the amount of information shown in the display. It enables you to setup a specific operator view for an operator. For example, you can set up a filter that shows only resources of a certain name or that are in a specific state. For more information refer to *System Automation for z/OS Operator's Commands*.

Commands that are Processed by the Automation Agents

```
AOFKSTA3          SA z/OS - Command Dialogs
Domain ID = IPUFM  ----- DISPSFLT -----      Date = 05/23/04
Operator ID = OPER1                               Time = 11:21:18

Resources      ( *, *esour*, resource or jobname )
AGFG AGFG2 APPC ASCH BATB000P100 BATB000P200 BATB000P300 BATB000S100
-----

Statuses      ( *, status or choice, leading ^ or \ negates)
*
-----

Preceded: 1 All resources that are currently UP or ENDED
          2 All resources that are in a normal status
          3 All resources that are in a down status
          4 All resources that are in a start transition state
          5 All resources that are in a stop transition state

Target Systems ( system name , domain ID , sysplex name or * )
AOC7
-----

Command ==>
PF1=Help  PF2=End   PF3=Return  PF4=Clear  PF5=Reset  PF6=Roll
PF9=Save  PF12=Retrieve
```

Figure 18. DISPFILT Command Dialog

The panel displays the filters that will be used for the DISPSTAT display. The filters are stored in task global variables, and you can update them by pressing PF9. When the panel is invoked from within DISPSTAT, pressing ENTER will take you to the DISPSTAT display, using the filters that are currently displayed. The filters are not saved unless you have previously pressed PF9 in the DISPSFLT command dialog.

The filters you can set are:

- **Resources:**

You can specify a list of one or more subsystem names. They can have leading or trailing wildcards (for example, *VIEW, NET* and *TVIE* are okay, but NE*EW will not work).

You can also specify a list of job names instead of subsystem names.

- **Statuses:**

You can specify a list of automation statuses to be included or excluded from the display. This filter is used whenever the display is refreshed.

SA z/OS provides the following sets of pre-grouped statuses, which can be abbreviated by their numbers:

1. All resources that are currently UP or ENDED
2. All resources that are in a normal status: DOWN, STARTED, UP, AUTOTERM, AUTODOWN, ACTIVE, ENDED, ENDING, RESTART, EXTSTART, RUNNING
3. All resources that are in a down status: DOWN, INACTIVE, RESTART, AUTODOWN, CTLDOWN, STOPPED, BROKEN, MOVED, FALLBACK
4. All resources that are in a start transition state: RESTART, STARTED, STARTED2, ACTIVE, RUNNING, EXTSTART
5. All resources that are in a stop transition state: AUTOTERM, ENDING, STOPPING, ABENDING, BREAKING

- **Target Systems:**

This lets you specify the systems within the sysplex that you want data gathered from. By default it shows just your local system.

Commands that are Processed by the Automation Agents

You can specify a list of one or more system and/or domain names and/or sysplex names, and the query is sent to those systems. Alternatively you can specify '*ALL', and the query is sent to all active systems within the sysplex.

In general the more specific your query and the fewer systems you send it to, the faster it will execute.

When you have updated the filters (the defaults will do for just exploring), press ENTER to go into the DISPSTAT display.

How to Use DISPINFO to Display Detailed Information

To see detailed information on a subsystem from the automation agent's point of view, use the DISPINFO command. This will only work on one subsystem at a time. If the subsystem you specify happens to be on more than one system within the sysplex, you will be prompted to select the one you would like to see information about.

The DISPINFO command displays information about the selected resource, in the following example, the APPC resource. As you can see, it displays a lot of the definitions that have been made to SA z/OS about the subsystem. You can use PF8 to scroll forward to see more information about this resource. You can use PF4 to toggle to the INGINFO command display for this resource which shows the automation manager view of it.

```
AOFKINFO          SA z/OS - Command Dialogs          Line 1   of 103
Domain ID   = IPUFM   ----- DISPINFO -----          Date = 06/18/04
Operator ID = OPER1                                     Time = 14:48:25

Subsystem ==>  APPC          System ==>  AOC7          System name, domain ID
                                                         or sysplex name
Subsystem   : APPC          on System : AOC7

Description  : APPC/MVS automation policy.

               The APPC/MVS address space must be operational
               before the Operations workstation can be
               fully functional. (In particular, I/O monitoring
               functions require the use of APPC/MVS.)

Class       : None
Job Name    : APPC
Job Type    : MVS

Automation status : UP
Last Monitored : 16:47:35 on 06/13/04
Last Changed  : 17:36:40 on 06/01/04
Last Message  :
               AOF571I 14:36:40 : APPC SUBSYSTEM STATUS FOR JOB APPC IS UP - AT
               SYS-OPS RECYCLE

Command ==>
PF1=Help    PF2=End      PF3=Return   PF4=INGINFO   PF6=Roll
PF8=Forward PF9=Refresh                                PF12=Retrieve
```

Figure 19. DISPINFO Command Dialog

Use PF8 to display the following:

Commands that are Processed by the Automation Agents

```
AOFKINFO          SA z/OS - Command Dialogs      Line 23  of 103
Domain ID = IPUFM  ----- DISPINFO -----      Date = 06/18/04
Operator ID = OPER1                               Time = 14:50:38

Subsystem ==>  APPC          System ==>  AOC7      System name, domain ID
                                                    or sysplex name

Monitor          : AOFMJMON (default)
Monitor Status   : ACTIVE

Monitor Interval : None specified

ARM Element Name : None

Association       : Primary

WLM Resource Name : None

Managing Feature  : None

Command Prefix    : None

Flags - Automation : Yes
      Init Start   : Yes
      Start        : Yes
      Recovery     : Yes
      Shut Down    : Yes
      Restart      : Yes

Command ==>
PF1=Help  PF2=End    PF3=Return  PF4=INGINFO      PF6=Roll
PF7=Back  PF8=Forward PF9=Refresh PF12=Retrieve
```

Use PF8 to display the following:

```
AOFKINFO          SA z/OS - Command Dialogs      Line 45  of 103
Domain ID = IPUFM  ----- DISPINFO -----      Date = 06/18/04
Operator ID = OPER1                               Time = 14:52:14

Subsystem ==>  APPC          System ==>  AOC7      System name, domain ID
                                                    or sysplex name

Restart after IPL : START
Restart after Recycle : START
Restart when Stopped : ABENDONLY

Thresholds -      APPC  SUBSYSTEM  DEFAULTS
Critical   : Undefined 2 in 01:00  5 in 01:00
Frequent   : Undefined 3 in 05:00  3 in 05:00
Infrequent : Undefined 4 in 12:00  1 in 10:00

Start Up Mechanism :
Scheduling Subsystem : Default

External startup   : NEVER

Prestart :
None

Startup :

Start command to be used :
MVS S APPC,APPC=07
```

Not all panels are shown here. Use PF8 until you reach a display line like the following, that indicates the end of the display:

Data produced at 14:52:35 on 06/18/02.

Commands that are Processed by the Automation Agents

```
AOFKINFO          SA z/OS - Command Dialogs          Line 111 of 143
Domain ID   = IPUFM          ----- DISPINFO -----          Date = 06/18/04
Operator ID = OPER1                                     Time = 14:52:38

Subsystem ==>  APPC          System ==>  AOC7          System name, domain ID
                                     or sysplex name

                ACTIVE - ACTIVE MESSAGE RECEIVED
2004-06-17 15:21:12 :
AAA006I APPC IS UP
2004-06-17 15:21:13 :
AOF571I 15:21:13 : APPC SUBSYSTEM STATUS FOR JOB APPC IS
UP - UP MESSAGE RECEIVED
AOF570I 15:21 : ISSUED "INGTIMER APPC EVERY
00:15:00,XX:33:00,17:00:00 PPT PIPE LIT /INGTIMER
INGPOSTSTART OF APPC/ | COLOR PINK | LOGTO NETLOG"
FOR SUBSYSTEM APPC - MSGTYPE IS POSTSTART

User Information
first line
second line

Configuration data has been last modified at 15:12:00 on 03/18/04

Data produced at 14:52:35 on 06/18/04.

Command ==>
PF1=Help    PF2=End      PF3=Return   PF4=INGINFO          PF6=Ro11
PF7=Back    PF9=Refresh          PF12=Retrieve
```

Note: If you have defined the exit routine AOFEXC03, the DISPINFO panel also shows user supplied information about the subsystem. Refer to *System Automation for z/OS Customizing and Programming* for more information about AOFEXC03.

If you use DISPINFO for a CICS- or IMS-controlled resource, then PF10 is active and with this you can view additional CICS or IMS related information as if you issued the INGCICS REQ=INFO or INGIMS REQ=INFO commands.

The following specific information is added for resources defined with INGVTAM startup and shutdown commands:

```
VTAM Communication Information :
Specific APPLID : IPSAMT          ACB      :
Generic APPLID  :                  USERVAR  :
Major Nodes    : IPSMBT
```

This displays the specific APPLID and optionally the ACB (if it is different from the APPLID) that the resource is using. It also shows the generic APPLID and USERVAR if any have been defined. Finally a list of major nodes that have been registered as being used by the resource is displayed.

The following specific information is displayed for a CICS resource:

```
CICS Information :
Version          : 6.1 (Transaction Server V2.1)
Indoubt UOW     : NO
Last Start Type : EMERGENCY
RLS Status      : RLSINACTIVE
XRF Status      : Active
```

Commands that are Processed by the Automation Agents

This displays the Version (both CICS and Transaction Server), the Indoubt Unit of Work status, the last start type and optionally the RLS and XRF status of the CICS system.

The following specific information is displayed for an IMS resource:

```
IMS Information :
Version          : 7.1
CQS Name        : None
FDR Name        : None
MVS sub-system ID : M712

IMS OLDS Information :
Minimum Available : 3      Spares          : 3
Active           : 1      Number         : 7
                   Archive
DDNAME  Status      Job Num.  Error
DFSOLP00 AVAILABLE
DFSOLS00 AVAILABLE
DFSOLP01 AVAILABLE
DFSOLS01 AVAILABLE
DFSOLP02 INUSE
DFSOLS02 INUSE
DFSOLP03 AVAILABLE
DFSOLS03 AVAILABLE
DFSOLP04 STOPPED          SPARE
DFSOLS04 STOPPED          SPARE
DFSOLP05 STOPPED          SPARE
DFSOLS05 STOPPED          SPARE
DFSOLP99 STOPPED          SPARE
DFSOLS99 STOPPED          SPARE
```

This shows the Version of IMS running, the subsystem name of the shared queues control task, the subsystem name of the Fast Database Recovery region, the command prefix for commands, the OLDS information with counts of active and spare OLDS and details of the status of each OLDS data set.

How to Use DISPTREE to Display Dependency Information

Generally if a resource has not been started, it will be because of a problem with one of its supporting resources. You may also wish to know which other subsystems will be impacted by problems with a particular subsystem. To find out what the problem is, use the DISPTREE command.

The DISPTREE command presents a view of the start dependencies, stop dependencies or the group membership of the specified subsystem (resource). Supporting resources are shown above and to the left of it, its dependent resources are shown below and to the right. The following screen shot shows the DISPTREE display for a start dependency for a JES2 subsystem. As you can see, JES2 in our example has no supporting resources.

Commands that are Processed by the Automation Agents

```
AOFKTREE          SA z/OS - Command Dialogs      Line 1   of 15
Domain ID   = IPUFM          ----- DISPTREE -----      Date = 03/13/04
Operator ID = OPER1                                     Time = 17:08:37

Subsystem ==> JES2          System ==> AOC7          Dependency ==> START

JES2

+-- IRRDPTAB
+-- JESWTR
+-- LOGRECJ
+-- RMF
!   +-- RMFIII
+-- SMFDUMPJ
+-- VTAM
    +-- APPC
        !   +-- ASCH
        +-- BNKTSO
            !   +-- BNKVTAM
            !       +-- BNKJES
        +-- TSO
        +-- UNIQUE

Command ==>
PF1=Help    PF2=End    PF3=Return    PF4=Focus    PF5=Info    PF6=Roll
PF9=Refresh PF11=Right PF12=Retrieve
```

| The actual screen uses reverse video to improve the highlighting. Non-subsystem
| resources are indicated with |.

| PF4 will refocus the display onto the subsystem under the cursor. PF5 will invoke
| DISPINFO against the subsystem under the cursor. However, you cannot use
| DISPINFO on non-subsystem resources.

How to Display Statistical Information

The INGRPT command displays a panel providing statistical information about the automation agent and some basic information about the automation manager of a system in a sysplex.

The summary report lists information about, for example:

- The number of resources defined to NetView and managed by the automation agent
- The total number of messages automated
- The total number of resulting commands
- The total number of START and STOP commands issued
- The number of systems in the sysplex
- The total number of resources managed by the automation manager

Commands that are Processed by the Automation Agents

```
INGKYRPO          SA z/OS - Command Dialogs      Line 1   of 45
Domain ID = IPSFM ----- INGRPT -----      Date = 04/12/04
Operator ID = OPER1          System = KEY1      Time = 17:45:36

***** Summary Statistics *****
System : KEY1                                     12 Apr 2004 17:45:34
-----

Begin of statistics : 04/08/04 07:52:00          Domain : IPSFM
End of statistics   : 04/12/04 17:45:34          Sysplex : KEY1PLEX
Elapsed time (hours) : 105:53 (6353 min)

Automation Agent Statistics
CPU time used (seconds) : 4428.44

Total number of resources defined : 83
Total number of resources managed : 16

Total number of messages automated : 13
Average number of messages per hour : 0.1

Total number of resulting commands : 7
Average number of commands per hour : 0.1

Total number of StartUp commands issued : 5
Total number of ShutDown commands issued : 0

Total number of workitems sent : 725
QUERY          569
UPDATER        156
Average number of workitems per hour : 13.1
Total number of TimeOuts : 0
Average wait time (in seconds) : 0.2
Maximum wait time (in seconds) : 0.4

Total number of orders received : 154
ACF             1
OBSERVER       115
OTHERS         1
START          37
Average number of orders per hour : 2.8

Automation Manager Statistics
Number of systems in sysplex : 2
Total number of resources : 233
Number of APL resources : 152
Number of APG resources : 48

Command ==>
PF1=Help      PF2=End      PF3=Return    PF6=Roll
PF8=Forward   PF9=Refresh   PF12=Retrieve
```

Figure 20. Statistics Summary — INGRPT Command Dialog

If you want to see detailed statistical information for a particular resource, issue the command INGRPT STATS=DETAIL. A panel similar to the following is displayed which shows statistical information about all resources that are automated by the automation agent.

Commands that are Processed by the Automation Agents

```

INGKYRP1                SA z/OS - Command Dialogs          Line 1 of 11
Domain ID = IPSFM      ----- INGRPT -----             Date = 04/12/04
Operator ID = OPER1    System = KEY1                      Time = 18:40:44

Cmd: S Show details

          ---Automated--- ---Commands---
Cmd  Name      Type System      Msgs  Cmts  Start  Stop  Crits
-----
EYUCMS1A  APL  KEY1          1     1     0     0     0
IMS711C4  APL  KEY1          2     2     0     0     0
IMS713C4  APL  KEY1          1     0     0     0     0
MHB       APL  KEY1          0     0     1     0     0
OPCF      APL  KEY1          1     2     0     0     0
PROC_OPS  APL  KEY1          0     0     1     0     0
SGG1MSTR  APL  KEY1          1     1     0     0     0
SG11      APL  KEY1          1     1     0     0     0
TESTAPPL  APL  KEY1          0     0     1     0     0
TWSCMDRVR APL  KEY1          0     0     1     0     0
TWSREQSRVR APL  KEY1          0     0     1     0     0

Command ==>
PF1=Help    PF2=End          PF3=Return      PF6=Roll
PF9=Refresh PF12=Retrieve
  
```

Figure 21. Statistical Information about Resources — INGRPT Command Dialog

For each resource the following information is shown:

- the number of messages automated
- the number of commands resulting from the message traps
- the number of START and STOP commands that were issued
- the number of critical threshold conditions that have occurred

Select a resource to display details by entering S in the Cmd column.

Commands that are Processed by the Automation Agents

```
INGKYRPO          SA z/OS - Command Dialogs      Line 1 of 22
Domain ID = IPSFM  ----- INGRPT -----      Date = 04/12/04
Operator ID = OPER1      System = KEY1          Time = 18:45:23

***** Detail Resource Statistics *****
System : KEY1                      23 Apr 2004 14:43:09
Resource : OPCF
-----

Begin of statistics : 04/23/04 09:35:32      Domain : IPSFM
End of statistics  : 04/23/04 14:43:09      Sysplex : KEY1PLEX
Elapsed time (hours) : 05:08 (308 min)

Total number of messages automated          : 1
Average number of messages per hour        : 0.2
Total number of resulting commands         : 2
Average number of resulting commands per hour : 0.4

Total number of StartUp commands issued    : 8
Total number of ShutDown commands issued   : 2

Number of critical threshold reached hits   : 1

Breakdown...                               message / command
POSTSTART                                  3      3
SHUTNORM                                   2      2
STARTUP                                    5      5

Command ==>
PF1=Help      PF2=End      PF3=Return      PF6=Roll
PF8=Forward   PF9=Refresh   PF12=Retrieve
```

Figure 22. Detailed Statistical Resource Information

Chapter 7. How to Monitor and Control Resources

This chapter introduces a series of commands that you can use to control resources. Controlling resources in this context means making them available or unavailable.

Besides the principal methods for controlling resources as introduced in Chapter 2, “Concepts,” on page 13, SA z/OS offers more sophisticated techniques to the operator.

With the commands introduced in this chapter, you can, for example:

- start or stop a resource or application group
- view resource or group details
- turn on or off the automation flag for a resource or application group
- view and modify service periods, referred to as schedules
- manage sysplex resources
- display resource statistics
- view trigger definitions

The following topics are discussed:

- “How to Start and Stop Resources”
- “How to Display Trigger Information—INGTRIG and DISPTRG” on page 99
- “How to Work with Application Groups” on page 107
- “How to Set a Timer” on page 112
- “How to Interact with the Automation Manager - INGAMS” on page 113
- “How to Change the Automation Agent Status - SETSTATE” on page 114
- “How to Monitor Hardware” on page 122
- “How to Monitor and Control a Sysplex” on page 141

How to Start and Stop Resources

Besides goal driven automation that you exploit by defining automation policy, SA z/OS offers a series of commands that let the operator react on special circumstances beyond regular automation. The commands for such purposes are introduced in the subsequent sections. Each command is documented with the complete syntax and some further sample output screens in *System Automation for z/OS Operator's Commands*.

How to Use INGREQ

As described in “Automation Concepts” on page 13, commands issued with INGREQ and a high priority are the most powerful automation commands to make applications available or unavailable. They supersede all other requests that exist against a resource at that time.

You can use this command to:

- initiate the start process of one or more specified application resources or application group resources

How to Start and Stop Resources

- initiate the shutdown process of one or more specified application resources or application group resources.

A request is persistent until:

- you cancel it using the INGVOTE command
- it expires according to the specified EXPIRE parameter
- it is automatically removed when a specified condition comes true
- you want it to be cancelled after a specified timeout period, if the request has not been satisfied after that time
- it is overwritten by a request from the same source

Even though requests are persistent, it is not guaranteed that they will be satisfied by the automation manager.

With a request you specify two categories of parameters:

- How should the request be executed (for example, with low or high priority)?
- What should happen with the request during its lifetime, for example, should it expire after a certain time anyway, or after a timeout if it has not been successful?

If you specify the command name without any parameters, SA z/OS displays a panel where you can specify, for example, startup or shutdown parameters, a priority, and a timeout for the request, as shown in Figure 23.

```
INGKYRU0                SA z/OS - Command Dialogs                Page 1 of 2
Domain ID = IPUFM        ----- INGREQ -----                Date = 05/21/04
Operator ID = OPER1                                           Time = 10:18:29

Resource => _____ format: name/type/system
System   => _____ System name, domain ID or sysplex name

Request  => _____ Request type (START, UP or STOP, DOWN)
Type     => NORM         Type of processing (NORM/IMMED/FORCE/user) or ?
Scope   => _____ Request scope (ONLY/CHILDREN/ALL)
Priority => LOW         Priority of request (FORCE/HIGH/LOW)
Expire   => _____ , _____ Expiration date(yyyy-mm-dd), time(hh:mm)
Timeout  => 0 / MSG   Interval in minutes / Option (MSG/CANCEL)
AutoRemove => _____ - see help for list
Restart  => NO         Restart resource after shutdown (YES/NO)
Override => NO         (ALL/NO/TRG/FLG/DPY/STS/UOW/INIT)
Verify   => YES         Check affected resources (YES/NO/WTOR)
Precheck => YES         Precheck for flags and passes (YES/NO)
Appl Parm => _____

AOF710A VERIFY/REVISE INPUT AND THEN PRESS ENTER
Command ==>>
PF1=Help    PF2=End    PF3=Return    PF6=Ro11
PF11=Next   PF12=Retrieve
```

Figure 23. INGREQ Command Dialog

With the *Request* parameter you specify whether you want to launch a request to make the specified resource available (startup) or unavailable (shutdown).

With the *Type* parameter you specify the type of the startup/shutdown. Valid shutdown types are NORM, IMMED and FORCE. Valid startup types are NORM and any other type defined by the user in the policy database for the resource. Specify a question mark (?) to see the list of user defined startup types.

Note: The value specified here will be overridden by the INGSET STARTTYPE/STOPTYPE value if previously specified. However, a stop type of FORCE, wherever specified will always be honored.

Use the VERIFY option to view the TYPE value that will be used for this request.

With the *Scope* parameter you can specify whether the startup/shutdown affects this resource and/or its descendants, as follows:

ONLY Only the resource is started/stopped.

ALL The resource and its dependants are started/stopped.

CHILDREN

Only the dependants are stopped. This option does not apply for a start request.

Note:

A request against a resource to make its children unavailable (MakeUnavailable, Scope=CHILDREN) will not affect the resource itself, as you do not want to make it unavailable, but only want to make the dependent resources unavailable. So this request is propagated to the dependent resources. On the original target resource, however, such a vote will never be considered for the winning vote evaluation. Whatever priority such a vote will have, it will never become the winning vote on this resource.

With the parameter *Precheck* you can specify whether or not SA z/OS should perform a validation check before the startup/shutdown command is issued. For example, SA z/OS would check if automation flags permit a startup/shutdown. YES is the default.

With the parameter *Appl Parm*s you can specify modifications and parameters to the START or STOP request that are particular to the application being started or stopped. This applies primarily to CICS, DB2 and IMS subsystems. The contents of this field is placed into the EHKVAR1 substitution variable. This variable can be used in the start or stop command.

With the parameter **MULT=YES** you can specify whether INGREQ will allow multiple resources to be targeted in line mode. Without this parameter the specification of multiple resources (by the use of wildcards) will be considered ambiguous and an error message will be issued.

Using PF11, you can specify descriptive information on the comment lines, for example, why the request was made, as shown in Figure 24 on page 92.

How to Start and Stop Resources

```
INGKYRU2          SA z/OS - Command Dialogs          Page 2 of 2
Domain ID = IPUFM  ----- INGREQ -----          Date = 05/21/04
Operator ID = OPER1                               Time = 10:18:47

Resource  => _____ format: name/type/system
System    => _____ System name, domain ID or sysplex name

Comment   => _____

Interrupt => NO Specify YES to terminate Startup processes
```

Figure 24. INGREQ Command Dialog

With the parameter *Interrupt* you can specify whether or not a request to shutdown a resource should interrupt the startup phase of a resource:

- YES** The automation manager interrupts the startup process as soon as this stop request is submitted. Interrupting the shutdown process means breaking the former stop request.
- NO** If the resource is starting, the automation manager waits for the resource's UP message before proceeding with the shutdown. This is the default.

After pressing ENTER to submit the request, you will see a verification panel that lists all resources that are affected either when posting an INGREQ command to the automation manager or when cancelling a previously made request. When making a request, a resource is affected if it is within the dependency graph that the request is propagated along. This will also display resources that are already in the desired state. In this case nothing will happen to the resource.

When cancelling a request, a resource is affected when it holds a vote for the request to be removed. Only resources that are highlighted will be processed by the command. All other resources are already in the state satisfying the appropriate dependencies.

With the *Verify* option you can decide whether you want to see a verification panel before submitting the request (see Figure 25 on page 93). Starting with the verification panel, you can manipulate affected requests, using the commands listed in the top of the panel, for example, you can invoke the INGVOTE command dialog for a selected resource. Or, you can go back to the INGREQ command dialog to change the options/parameters.

In the *TRG* column of the INGREQ verification dialog form Figure 25 on page 93, you can see whether for a resource a trigger condition is satisfied.

In the *SVP* column, you can see whether a service period is linked to the resource and whether the current time is within the service period.


```

AOFKVFY1          SA z/OS - Command Dialogs          Line 1 of 3
Domain ID = IPUFM ----- INGREQ -----          Date = 03/13/04
Operator ID = OPER1                               Time = 18:05:21

Verify list of affected resources for request START

CMD: S show overrides  T show trigger details  V show votes
Cmd Name      Type System  TRG SVP  W Action Type  Observed Stat
-----
-  APPC        APL  AOC7          Y      NORM  AVAILABLE
-  JES2        APL  AOC7          NORM  AVAILABLE
-  VTAM        APL  AOC7          NORM  AVAILABLE
-

```

Figure 25. INGREQ Verification

The **W** column indicates whether this vote is the winning vote for this resource.

Note: You can use a DB2 specific feature for shutdown on the INGREQ panel: In the case of using the *Precheck* option on the INGREQ command dialog, this will check the DB2 subsystem for INDOUBT threads. If there are any, then the ING107E message is issued and the shutdown disallowed. However, you can ask for this precheck not to be done by entering Precheck=N. Later when the actual subsystem is asked to be shutdown, the INDOUBT check is again performed although the Precheck was set no N. To get over this, you can enter an override of ALL or UOW and the shutdown will go ahead without this check.

Starting Subsystems

In order to start a subsystem it must be put into either the DOWN or the RESTART automation status seen by the automation agent. The DOWN status occurs just after SA z/OS initialization and indicates that the subsystem has not been active since SA z/OS was last started.

To put a subsystem into the RESTART status, you can use the SETSTATE command.

If Subsystems Do not Start

If you have asked for a subsystem to start and SA z/OS refuses to do it, there are some things you can check.

At first examine the AOF313I message. This should clearly state why the startup request was rejected. If there isn't one, then search your netlog for more information.

- *Supporting Resources Not Up*

If supporting resources are not up, SA z/OS tries to start them. If you wait a while they should come up and then the target subsystem will be started.

You can check on the state of the parents through the DISPTREE display, including status, or through messages issued to the NCCF console. If for example, it turns out that a parent is in a status of BROKEN, it is not possible to start the desired subsystem until this problem is fixed.

- *Automation Flags Turned Off*

Automation flags may be turned off for the subsystem or for just the INITSTART or RESTART actions. If the INITSTART flag is turned off, the subsystem cannot be started from a DOWN status. If the RESTART flag is turned off, the subsystem cannot be started from a RESTART status.

How to Start and Stop Resources

To see the flag settings for a subsystem you can use either DISPFLGS or DISPINFO. DISPFLGS will show you both the current settings in force for the subsystem and, perhaps more importantly, the settings that were made which resulted in those values.

- ***Trigger Not Satisfied***

Triggers tell SA z/OS when it is allowed to stop and start subsystems. If a trigger has been linked to a subsystem, SA z/OS will not stop or start that subsystem unless the trigger says it is OK to do so. Normally this function is fine, but there may be times when you need to override it.

To find out about the trigger linked to a subsystem, use the DISPTRG command. For example, the command

```
DISPTRG TEST0,TARGET=KEY3
```

shows the triggers and events associated with the resource *TEST0*.

If you want to start something and override the event settings you can either issue an INGREQ REQ=START OVERRIDE=TRG command or with the INGREQ command, you can specify that trigger conditions should not be considered by specifying option *Override ==> TRG* in the panel from Figure 23 on page 90. Or if you want to ensure that the trigger is satisfied before you start the resource, you can set it with INGEVENT. INGEVENT can be used to set an event (but may affect more than just the subsystem you are dealing with).

- ***The Automation Manager Automation Flag Is Off:***

Use the INGINFO command or the INGLIST command to find out what the current setting is. If the automation flag is off, the automation manager will not send the order to start the subsystem to the automation agent.

For a detailed scenario on how to use SA z/OS diagnosis methods, refer to Chapter 8, "Scenario: Finding Out Why a Resource Is not Available," on page 191.

If Subsystems Do not Stop

If you have asked for a subsystem to stop and SA z/OS refuses to do so, here are some things you can check:

- ***supporting resource cannot stop:*** the supporting resources cannot be brought in the condition fulfilling the stop dependency, so that the dependent resource that you want to stop, also cannot stop. This is the case if the compound status indicates: AWAITING.
- ***request with higher priority to have the subsystem UP:***
Use the INGVOTE command to determine if a higher priority request exists for the subsystem. If so, you can either reissue the stop request with a higher priority thus overruling the other request or cancel the other request.
- ***the shutdown trigger is not satisfied:***
Use the DISPTRG command to find out the trigger conditions defined for the subsystem. Watch the shutdown conditions that are defined for the subsystem. You can either set one or more events bringing the shutdown condition into a satisfied state or use the INGREQ OVERRIDE=TRG parameter to bypass the trigger check.
- ***automation flags turned off:***
The automation flag may be turned off for the subsystem or just the terminate action. To see the flag settings, use DISPSTAT, DISPFLGS, or DISPINFO.
- ***the automation manager automation flag is off:***

Use the INGINFO command or the INGLIST command to find out what the current setting is. If the automation flag is off, the automation manager will not send the order to stop the subsystem to the automation agent.

- *check history:*

Use the INGINFO command to analyze the history of the resource.

How to Use INGVOTE

Whenever you (or a program) want the automation manager to do something, you must send it a request. Requests are remembered within the automation manager. They are made against individual resources and then propagated over the relationship graph to other resources within the automation manager. If for example, your request is to start resource A, and A has a relationship to B in a way that B needs to be started before A can run, then the automation manager generates a vote to resource B to start it. So a request can be viewed as a black box of related votes generated by the automation manager.

If you enter the INGVOTE command without any parameter, it displays all pending requests that are persistent at the automation manager.

If you enter the INGVOTE command specifying a resource, or launch it from the INGLIST command dialog, then SA z/OS displays all pending votes for this given resource with the following information:

- name of resource the request was made against
- request action
- source of request
- request priority
- request vote

```

INGKYRQ2                               SA z/OS - Command Dialogs           Line 1 of 15
Domain ID = IPUFM                       ----- INGVOTE -----           Date = 06/27/04
Operator ID = OPER1                     Sysplex = AOCPLEX                   Time = 14:12:47

Cmd: C Cancel request   K Kill request   S Show details   V Show votes
Cmd Name      Type System      Request Data
-----
_  SVPEVENT    SVP              Req : MakeUnAvailble
                   At : 2004-06-27 10:00:01
                   Org : SCHEDULE
                   Pri : 01120000  Should Be Down - Schedule
                   Stat: Winning
_  SVPH0730    SVP              Req : MakeUnAvailble
                   At : 2004-06-27 08:00:00
                   Org : SCHEDULE
                   Pri : 02120000  Must Be Down - Schedule
                   Stat: Winning
_  SVPL1200    SVP              Req : MakeUnAvailble
                   At : 2004-06-27 12:30:00
                   Org : SCHEDULE

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Roll
PF8=Forward   PF9=Refresh  PF12=Retrieve
    
```

Figure 26. INGVOTE Command Dialog

From the INGVOTE command dialog, you can enter the following commands against a selected vote:

- C** Cancels the request. This command displays the verification panel that lists

How to Start and Stop Resources

all resources that are affected if you want to cancel the selected vote/request (see also “How to Use INGREQ” on page 89).

- K** Cancels the request without showing the verification panel.
- S** Shows details about the request.
- V** Displays all the requests/votes that have been entered for the selected resource.

How to Use INGHIST

The INGHIST command is a sysplex-wide command issued against the sysplex’s automation manager. It displays history information about the work items processed by the automation manager. You can specify a certain resource for which you want SA z/OS to display the work history, or you can specify a time interval, for which you want to see all activities or you can combine both criteria.

Time-interval search criteria are entered in local time. However, because the history data is stored using GMT time stamps, it is necessary for INGHIST to convert the local-time search criteria to GMT before retrieving the data. This may make it necessary for the operator to make a manual adjustment to the time-interval search criteria after a daylight-saving-time change has been set. The local-time-to-GMT conversion calculation will be one hour different after a daylight-saving-time change. Data written after a daylight-saving-time change will be retrieved as expected but data written before will be offset by one hour.

If you enter this command without parameters, you receive the output for the complete sysplex starting from the last hour before you entered this command. Your output panel may look similar to the following:

```
INGKYHIO                SA z/OS - Command Dialogs          Line 1 of 112
Domain ID = IPUFM        ----- INGHIST -----          Date = 05/23/04
Operator ID = OPER1      Sysplex = AOC7PLEX                Time = 16:46:17

Start time ==> 2004-05-23 15:46:00    format: yyyy-mm-dd hh:mm:ss
End time   ==>                               Resource ==>

Cmd: + Expand workitem (e)  - Collapse workitem (c)
Cmd Date      Time      Footprint
-----
- 2004-05-23 16:46:14 GLOBAL automation flag is set off
- 2004-05-23 16:46:05 INITSTART automation flag is set off
- 2004-05-23 16:44:21 INITSTART automation flag is set off
- 2004-05-23 16:44:12 GLOBAL automation flag is set off
- 2004-05-23 16:43:32 GLOBAL automation flag is set off
- 2004-05-23 16:43:23 INITSTART automation flag is set off
- 2004-05-23 16:42:11 GLOBAL automation flag is set off
- 2004-05-23 16:42:02 INITSTART automation flag is set off
- 2004-05-23 16:40:24 INITSTART automation flag is set off
- 2004-05-23 16:40:14 GLOBAL automation flag is set off
- 2004-05-23 16:23:15 Startup for PROC_OPS/APL/AOC7 completed

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Roll
               PF8=Forward  PF9=Refresh  PF12=Retrieve
```

In the *Cmd* column on the left hand side of the INGHIST panel, with the + or the *e* command, you can expand each workitem so that you see information about which detail actions have been performed by the automation manager and which resources have been involved. In Figure 27 on page 97 you see for example, that at first the automation status has been set to *Idle* for resource *PROC_OPS/APL/AOC7*.

```

2004-05-23 16:23:15 Startup for PROC_OPS/APL/AOC7 completed
HSAL6269I Status/Automation is Idle (PROC_OPS/APL/AOC7)
HSAL6282I Status/Compound is Satisfactory (PROC_OPS/APL/AOC7)
HSAL6172I Group Observer update sent (PROC_OPS/APL/AOC7)
HSAL6157I Make Available order completed (PROC_OPS/APL/AOC7)
    
```

Figure 27. Expanded Workitem

With - or c, you can collapse expanded work items again.

How to Use Schedules—INGSCHED

Service periods (also called schedules) are one of the methods to set goals for resources. Service periods are defined as permanent automation policy in the customization dialog. To display all service periods defined in the sysplex, use the INGSCHED command dialog as shown in Figure 28.

```

INGKYSPO          SA z/OS - Command Dialogs          Line 1 of 7
Domain ID = IPUFM  ----- INGSCHED -----         Date = 03/14/04
Operator ID = OPER1      System = AOC7              Time = 17:13:58

Cmd:  A Show Details  B Show Overrides  C Show Resources

Cmd Schedule Description
-----
- BZOETEST  SVP with a lot of entries
- KRZTEST
- SHIFT1   Weekday A.M. (06:00 - 11:59)
- SHIFT2   Weekday P.M. (13.00 - 16.59)
- a SHIFT3   Mo-Fr 16.00 - 05.55 ... 16.04 - 05.59
- SHIFT4   Saturday 6.00 - 17.59
- SHIFT5   Sunday 6.00 - 12.00
    
```

Figure 28. INGSCHED Command Dialog

On the INGSCHED panel, you can enter the following command codes for a selected schedule:

- A** Using this action code, you can see the UP and DOWN time windows of the selected service period for the week as shown in Figure 29 on page 98 for service period SHIFT3 in our example. No updates are possible on this panel.
- B** Displays the actual UP and DOWN time windows for the selected schedule including the overrides starting with the current day. Use this option if you want to override a schedule for all resources that use this schedule (that means, for all resources, to which this schedule is linked in the customization dialog). Such an override is called a *schedule override* and is described in “Schedule Override” on page 98.
- C** Displays the resources that are using the selected schedule. You can select a resource to view the schedule that is used for this resource. Now you can override the schedule just for the selected resource. Such an override is called a *resource override* and is described in “Resource Override” on page 99.

How to Start and Stop Resources

```

INGKYSP1          SA z/OS - Command Dialogs          Line 1 of 14
Domain ID = IPUFM  ----- INGSCHED -----          Date = 03/14/04
Operator ID = OPER1      System = AOC7              Time = 17:17:03

Schedule. . . : SHIFT3          Mo-Fr 16.00 - 05.55 ... 16.04 - 05.59

  Day      Pri  From-To  From-To  From-To  From-To  From-To
  ---      -  - - - -  - - - -  - - - -  - - - -  - - - -
MON  UP  H  1600-2359  - - - -  - - - -  - - - -  - - - -
    DN  H  0000-1559  - - - -  - - - -  - - - -  - - - -
TUE  UP  H  0000-0555  1601-2359  - - - -  - - - -  - - - -
    DN  H  0556-1600  - - - -  - - - -  - - - -  - - - -
WED  UP  H  0000-0556  1602-2359  - - - -  - - - -  - - - -
    DN  H  0556-1601  - - - -  - - - -  - - - -  - - - -
THU  UP  H  0000-0557  1603-2359  - - - -  - - - -  - - - -
    DN  H  0557-1602  - - - -  - - - -  - - - -  - - - -
FRI  UP  H  0000-0558  1604-2359  - - - -  - - - -  - - - -
    DN  H  0559-1603  - - - -  - - - -  - - - -  - - - -
SAT  UP  H  0000-0559  - - - -  - - - -  - - - -  - - - -
    DN  H  0600-2400  - - - -  - - - -  - - - -  - - - -
SUN  UP  L  - - - -  - - - -  - - - -  - - - -  - - - -
    DN  H  0000-2359  - - - -  - - - -  - - - -  - - - -
  
```

Figure 29. INGSCHED Details

The column with the **UP** and **DN** entries indicates whether the schedule is an UP or DOWN time window.

Schedule Override

You can interactively override a schedule for all resources that use this schedule. You can interactively override the schedule starting from a specified starting date for selected dates in the near or far future.

To reach the panel shown in Figure 30, select option **B** from the panel shown in Figure 28 on page 97 or use the INGSCHED command with the schedule name as a parameter. Schedules that have been overwritten are shown in yellow. From this panel you can now overtype or complete the displayed schedule.

```

INGKYSP2          SA z/OS - Command Dialogs          Date = 05/23/04
Domain ID = IPUFM  ----- INGSCHED -----          Time = 18:12:22
Operator ID = OPER1      Sysplex = AOC7PLEX

Schedule . . . : SHIFT3          Daily 16.00 - 19.00 high UP
Starting date => 05 / 23 / 2004    (mm/dd/yyyy)

C Date          Pri  From-To  From-To  From-To  From-To  From-To
- - - - -      -  - - - -  - - - -  - - - -  - - - -  - - - -
05/23/04 Tue  UP  H  1600 1900  - - - -  - - - -  - - - -  - - - -
                DN  H  0000 1600  - - - -  - - - -  - - - -  - - - -
05/24/04 Wed  UP  H  1600 1900  - - - -  - - - -  - - - -  - - - -
                DN  H  0000 1600  - - - -  - - - -  - - - -  - - - -
05/25/04 Thu  UP  H  1600 1900  - - - -  - - - -  - - - -  - - - -
                DN  H  0000 1600  - - - -  - - - -  - - - -  - - - -
05/26/04 Fri  UP  H  1600 1900  - - - -  - - - -  - - - -  - - - -
                DN  H  0000 1600  - - - -  - - - -  - - - -  - - - -
05/27/04 Sat  UP  H  1600 1900  - - - -  - - - -  - - - -  - - - -
                DN  H  0000 1600  - - - -  - - - -  - - - -  - - - -

Command ==>>
PF1=Help      PF2=End      PF3=Return    PF6=Ro11
PF7=Backward  PF8=Forward  PF9=Refresh   PF12=Retrieve
  
```

Figure 30. Panel for Schedule Overrides

Resource Override

To reach the panel shown in Figure 31, select option **C** from the panel shown in Figure 28 on page 97 and then select the resource for which you want to override the schedule.

Another way to reach the resource override panel is to issue the **INGSCHED** command with a certain resource name as a parameter, like:

```
INGSCHED AGFG2/APL/AOCA
```

Finally, there is a third way to reach the resource override panel: from the **INGLIST** command dialog (see “How to Use **INGLIST** to View Resources” on page 71) you can issue action code **I** against a selected resource. In the upcoming panel you will see the desired schedule. Service windows that are overrides for the resource are shown in pink. Now you can edit the schedule by typing new values.

```

INGKYSP2                SA z/OS - Command Dialogs
Domain ID = IPUFM      ----- INGSCHED -----      Date = 05/23/04
Operator ID = OPER1    Sysplex = AOC7PLEX              Time = 18:34:32

Resource . . . : BZOLGSVP/APL/AOC7          Schedule . . . : BZOETEST
Starting date => 05 / 23 / 2004      (mm/dd/yyyy)

C Date          Pri  From-To    From-To    From-To    From-To    From-To
-----
05/23/04 Tue  UP  H  0800 0810  _____
                                DN  H  0000 0759  _____
05/24/04 Wed  UP  L  0600 0700  _____
                                DN  L  0000 0600  _____
05/25/04 Thu  UP  H  1200 1600  _____
                                DN  H  0000 1159  _____
05/26/04 Fri  UP  H  1200 1230  _____
                                DN  H  0000 1159  _____
05/27/04 Sat  UP  L  0900 1000  _____
                                DN  L  0000 0859  1000 2400  _____

Command ==>
PF1=Help      PF2=End      PF3=Return    PF6=Ro11
PF7=Backward  PF8=Forward  PF9=Refresh   PF12=Retrieve
    
```

Figure 31. Panel for Resource Overrides

From this panel which is specific for the current resource you can:

- enter new service times
- overtype or blank out existing service times
- and delete a complete (pink) schedule override by entering action code **D** into the command column. Then the schedule override will disappear and the general schedule for all resources will be restored for the current resource (in green).

How to Display Trigger Information—INGTRIG and DISPTRG

Triggers are a very powerful tool of goal driven automation. As they influence the behavior of applications and application groups, the automation programmer needs to be very careful in connecting triggers to resources. For information on event and trigger support refer to *System Automation for z/OS Defining Automation Policy*.

In SA z/OS two commands are available to display trigger information:

How to Start and Stop Resources

INGTRIG

This command shows all triggers that are defined and used by resources within the automation manager scope. For each trigger you can view all resources that use this trigger.

```
INGKYTR0          SA z/OS - Command Dialogs      Line 1   of 13
Domain ID   = IPUFM      ----- INGTRIG -----      Date = 03/23/04
Operator ID = OPER1          Sysplex = AOCPLEX          Time = 16:31:20

Cmd: S show associated resources

Cmd Trigger  Description
-----
- BATCHSTP  Batch jobs B000P100,P200,P300
- BATCHSTR  Batch jobs B000S100,S200,S300
- DRWTRG1   TD Trigger 1 (evt1 - evt8)
- DRWTRG2   TD trigger 2 (evt9 - evt12)
- HUGTRIG1  Trigger for HUG applications
- MSCTRG1   scheduling test trigger 1
- MSCTRG2   scheduling test trigger 2
- TRIGGER6  Hourly window + batch controlled events
- TRIGGER7  Command controlled events, No UNSET
- TRIGGER8  Command controlled events, diff. unsets

Command ==>
PF1=Help    PF2=End    PF3=Return  PF6=Roll
             PF9=Refresh PF12=Retrieve
```

DISPTRG

This command displays the trigger conditions for a resource. To display a panel similar to the following, enter:

```
DISPTRG BATB000P100
```

(If you enter this command without parameters, SA z/OS prompts you for more information).

```
AOFLT000          SA z/OS - Command Dialogs      Line 1   of 1
Domain ID   = IPUFM      ----- DISPTRG -----      Date = 05/23/04
Operator ID = OPER1          System = AOC7          Time = 18:41:22

Resource ==>  BATB000P100/APL/AOC7
System ==>          System name, domain id or sysplex name
Trigger. . . : BATCHSTP          Observed status : HARDDOWN

Cmd: S show details
Cmd Type  Events
-----
- STARTUP  BATCHSTP

Command ==>
PF1=Help    PF2=End    PF3=Return  PF6=Roll
             PF9=Refresh PF12=Retrieve
```

With the command code *S*, for each trigger you can view the events that must be set to activate the trigger. You will see a panel similar to the following one:


```

AOFLT100          SA z/OS - Command Dialogs      Line 1   of 1
Domain ID   = IPUFM    --- Trigger Condition List ---- Date = 05/24/04
Operator ID = OPER1      System = AOC7           Time = 16:48:32

Trigger . . . . : BATCHSTP
Condition type. : STARTUP
Resource. . . . : BATB000P100/APL/AOC7          Observed status : HARDDOWN

Cmd: D show resources  S set event for resource  U unset event for resource
Cmd Event   Status    Unset  Description
-----
_  BATCHSTP  UNSET   NONE   Batch jobs B000P100,P200,P300

Command ==>
PF1=Help    PF2=End      PF3=Return   PF6=Ro11
PF9=Refresh PF12=Retrieve
    
```

Figure 32. Trigger Condition List

Note that in our example, the trigger and its event have the same name *BATCHSTP*. The event for this trigger is currently unset. With the command code *S*, you can set it.

How to Display Event Information—DISPEVT, DISPEVTS, and INGEVENT

You can reach the DISPEVT command dialog by issuing DISPEVT *event_name* or using the command code *D* from the trigger condition list shown in Figure 32.

This panel shows all resources that have the specified event defined within their trigger definition. For each resource the following information is displayed:

- the name of the resource
- the type of the resource
- the name of the system where the resource resides
- the name of the trigger that the resource is linked to.
- the status of the event (SET or UNSET). If the *Status* column remains blank this means that the status is unknown. This is treated as UNSET.

How to Start and Stop Resources

```

AOFLE000          SA z/OS - Command Dialogs          Line 1 of 3
Domain ID = IPUFM  ----- DISPEVT -----          Date = 05/24/04
Operator ID = OPER1          Sysplex = AOC7PLEX          Time = 16:43:21

Event      ==> BATCHSTP
System     ==> AOC7          System name, domain id or sysplex name
Description . . : Batch jobs B000P100,P200,P300
Unset condition. : NONE

Resource   Type  System  Trigger  Status
-----
BATB000P100  APL  AOC7    BATCHSTP  SET
BATB000P200  APL  AOC7    BATCHSTP  SET
BATB000P300  APL  AOC7    BATCHSTP

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Roll
               PF9=Refresh   PF12=Retrieve
  
```

Figure 33. DISPEVT Command Dialog

With the INGEVENT command, you can set or unset an event for a specified resource or for all resources in the sysplex for which it is defined.

To reach the INGEVENT command dialog, perform the following steps:

1. Issue the DISPEVTS command from any command line. You will see a panel as shown in Figure 34 which displays the events defined for resources in the sysplex.

```

AOFLE100          SA z/OS - Command Dialogs          Line 1 of 29
Domain ID = IPUFM  ----- DISPEVTS -----          Date = 05/31/04
Operator ID = OPER1          Sysplex = AOC7PLEX          Time = 14:41:46

Cmd: D show related resources      S set/reset event

Cmd Event   Unset  Description
-----
BATCHSTP   NONE   Batch jobs B000P100,P200,P300
BATCHSTR   NONE   Batch jobs B000S100,S200,S300
B000P100   DOWN   Stop cond 1 Trigger6 (batch job end)
B000P200   DOWN   = P1
B000P300   DOWN   Stop cond 2 Trigger6 (batch job 1 min)
B000S100   UP     Start cond 1 Trigger6 batch job end
B000S200   UP     = P1
B000S300   UP     Start cond 2 Trigger 6 batch end (1 min)
MDOEVENT   NONE   Donies Event1
MDOEVT2    DOWN   Donies Event2
MDOEVT3    START  Donies Event3
MDOEVT4    UP     Donies Event4

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Roll
               PF8=Forward   PF9=Refresh   PF12=Retrieve
  
```

Figure 34. DISPEVTS Command Dialog

From this command dialog, you have the following options:

- D** show related resources
- S** set/reset event

- Now use option *S* for a selected event to reach the INGEVENT command dialog shown in Figure 35.

Note: With the line mode INGEVENT command, you can directly set or unset an event for all or a specified resource, like for example:

```
INGEVENT BATCHSTP REQ=UNSET RESOURCE=BATB000P200/APL/AOC7
```

After successful completion, this will produce the message:

```
AOF442I UNSET EVENT BATCHSTP DONE FOR RESOURCE BATB000P200/APL/AOC7
ON AOC7. - FROM=AOC7
AOF099I FUNCTION COMPLETED
```

```
AOFLE110          SA z/OS - Command Dialogs
Domain ID   = IPUFM          ----- INGEVENT -----      Date = 05/31/04
Operator ID = OPER1          Sysplex = AOC7PLEX              Time = 15:34:04

Event . . . . . : BATCHSTP
Description . . . : Batch jobs B000P100,P200,P300
Unset condition . . : NONE

Specify/revise the following parameters and then press enter:

Function ==> UNSET          SET or UNSET
Resource ==> BATB000P100/APL/AOC7      Resource name

System ==> AOC7            System name, domain ID or sysplex name

AOF442I UNSET EVENT BATCHSTP DONE FOR RESOURCE BATB000P100/APL/AOC7 ON AOC7.
Command ==>
F1=Help      F2=End      F3=Return      F6=Ro11
F12=Retrieve
```

Figure 35. INGEVENT Command Dialog

Determining Automation Agent Flag Values

You can determine the current values of the automation flags for a particular subsystem using the DISPSTAT command dialog.

The DISPSTAT command dialog also provides a fast path to other command dialogs so you can change certain automation policy settings for a particular subsystem. It also shows the status of each subsystem.

How to Start and Stop Resources

```

A0FKSTA5          SA z/OS - Command Dialogs          Line 1 of 442
Domain ID = IPSFM ----- DISPSTAT -----          Date = 04/17/04
Operator ID = OPER1                                     Time = 14:24:01
A ingauto B setstate C ingreq-stop D thresholds E explain F info G tree
H trigger I service J all children K children L all parents M parents
CMD RESOURCE      STATUS      SYSTEM      JOB NAME      A I S R D RS TYPE      Activity
-----
- AGFG             BROKEN      KEY3        AGFG          Y Y Y Y Y Y MVS      --none--
- AGFG2            BROKEN      KEY3        AGFG2         Y Y Y Y Y Y MVS      --none--
- ANELSTP1         UP          KEY3        ANELSTP1      Y Y Y Y Y Y MVS      STOPPING
- ANELTEST         UP          KEY3        ANELTEST      Y Y Y Y Y Y MVS      STOPPING
- ANEL0AA1         UP          KEY3        ANEL0AA1      Y Y Y Y Y Y MVS      --none--
- ANEL00A1         UP          KEY3        ANEL00A1      Y Y Y Y Y Y MVS      STOPPING
- ANEL00B1         UP          KEY3        ANEL00B1      Y Y Y Y Y Y MVS      STOPPING
- ANEL00B2         UP          KEY3        ANEL00B2      Y Y Y Y Y Y MVS      --none--
- ANEL00B3         UP          KEY3        ANEL00B3      Y Y Y Y Y Y MVS      --none--
- ANEL00B4         UP          KEY3        ANEL00B4      Y Y Y Y Y Y MVS      --none--
- ANEL00B5         UP          KEY3        ANEL00B5      Y Y Y Y Y Y MVS      --none--
- ANEL00C1         UP          KEY3        ANEL00C1      Y Y Y Y Y Y MVS      --none--
- APPC             UP          KEY3        APPC          Y Y Y Y Y Y MVS      --none--
- ASCH             UP          KEY3        ASCH          Y Y Y Y Y Y MVS      --none--
- BATB000P100     DOWN       KEY3        B000P100     Y Y Y Y Y Y TRANS  --none--
- BATB000P200     DOWN       KEY3        B000P200     Y Y Y Y Y Y TRANS  --none--
- BATB000P300     DOWN       KEY3        B000P300     Y Y Y Y Y Y TRANS  --none--
- BATB000S100     DOWN       KEY3        B000S100     Y Y Y Y Y Y TRANS  --none--
- BATB000S200     DOWN       KEY3        B000S200     Y Y Y Y Y Y TRANS  --none--
- BATB000S300     DOWN       KEY3        B000S300     Y Y Y Y Y Y TRANS  --none--
- BLSJPRMI        ENDED      KEY3        BLSJPRMI     Y Y Y Y Y Y ONE SHOT --none--

Command ==>
PF1=Help  PF2=End    PF3=Return  PF4=INGLIST  PF5=Filters  PF6=Ro11
          PF8=Forward PF9=Refresh PF10=Previous PF11=Next   PF12=Retrieve

```

Figure 36. Example Display of Status of all Automated Resources (DISPSTAT)

The automation flags are:

- A** Automation
- I** Initial Start
- S** Start
- R** Recovery
- D** Shutdown
- R** Restart

Automation flags are set by the automation programmer in the customization dialog. Their meanings are explained in detail in *System Automation for z/OS Defining Automation Policy*.

Each flag has one of the following values:

- there is no explicit setting.
- N** the flag is turned off.
- E** The value of the automation flag depends upon the values returned by one or more user exit and will be determined when the value is required.
- Y** the flag is turned on.
- ?** there is an error.

Displaying Actual and Effective Automation Agent Flags

In addition to using the DISPSTAT command dialog to see the current automation flags for every subsystem, you can use the DISPFLGS command dialog to display both the actual flags that are set, and the effective flags for those subsystems that have explicit flags coded.

To display the actual and effective automation flags type DISPFLGS on any command line.

```

AOFKAAU                SA z/OS - Command Dialogs                Line 1    of 123
Domain ID = IPSFM      ----- DISPFLGS -----                Date = 04/30/04
Operator ID = OPER1                                         Time = 11:00:35

Cmd:  A Add flags    C Change flags    R Reset flags    S Scheduled Overrides
      Actual         Effective
Cmd  System  Resource  A I S R D RS  A I S R D RS
---  -
KEY1  APPC
KEY1  ASCH
KEY1  CICS1G
KEY1  AUTO
KEY1  SOS
KEY1  TRAN
KEY1  VIOL
KEY1  VTAMACB
KEY1  CICS1H
KEY1  AUTO
KEY1  SOS
KEY1  TRAN

Command ==>
PF1=Help      PF2=End      PF3=Return    PF6=Ro11
PF8=Forward   PF9=Refresh  PF12=Retrieve
    
```

Figure 37. DISPFLGS Command Dialog Panel

Two sets of flags are shown. Those on the left of the panel are the actual flags, and those on the right are the effective flags. The values shown for each flag are explained in “Determining Automation Agent Flag Values” on page 103.

The following command codes are supported:

- A** allows you to define automation flags for a resource using the same flag settings as the selected resource. Another panel is displayed where you can specify the resource name and optionally overwrite the flag settings.
- C** allows you to modify the automation flags of the selected resource. Another panel is displayed showing the current flag settings where you can overwrite the flag values.
- R** causes to reset the automation flags to the values specified in the automation control file (ACF).
- S** shows the scheduled override settings of the automation flags for the selected resource. You can specify scheduled overrides using the customization dialog. These are specific times when the automation is turned off for a particular flag and resource.

Displaying Automation Agent Flag Override Settings

The customization dialog allows you to turn automation flags off for a scheduled time period. You can view these time period settings by issuing the DISPSCHD command.

How to Start and Stop Resources

```
AOFKAASC          SA z/OS - Command Dialogs          Line 1 of 1
Domain ID = IPUFM ----- DISPSCHD -----          Date = 05/23/04
Operator ID = OPER1                                     Time = 09:13:10

System  Resource  Fl Day      Start - End      Flags
-----  -
KEY3    VTAM           S MONDAY  08:00:00 - 10:00:00  A Automation
KEY3    TSO            I SUNDAY   00:00:00 - 01:00:00  I Initial Start
                                           S Start up
                                           R Recovery
                                           D Shut down
                                           RS Restart

Command ==>
          PF1=Help   PF2=End   PF3=Return
          PF6=Roll   PF8=Forward PF9=Refresh PF12=Retrieve
```

- The System field shows the name of the system where the resource is defined.
- The Resource field shows the name of the resource.
- The Fl (flag) field shows an abbreviation of the automation flag name.
- The Day field shows the day of the week that automation is turned off.
- The Start and End fields show the time periods that automation is turned off.
- The Flags field shows the full name and the one character abbreviation for each automation flag.

In our example, Startup is not automated for VTAM on Mondays from 8 to 10 pm and Initial start is not automated for TSO on Sundays from midnight to 1 am.

Setting Automation On or Off—INGAUTO

You can change the automation setting for a specific automated resource or a group of automated resources using the INGAUTO command dialog. You can turn all automation off or only particular automation flags.

Note: Be sure to specify the correct subcomponent name. Although the subcomponent does not exist, the message FUNCTION COMPLETED will appear.

If you type INGAUTO on any command line, you see the following panel:

```

AOFKAASF          SA z/OS - Command Dialogs
Domain ID = IPUFM  ----- INGAUTO -----      Date = 05/23/04
Operator ID = OPER1                               Time = 09:13:10

Resource name => TSO_____ Specific or group resource or DEFAULT
Sub-components => _____ Resource sub-component
Note : Be sure to specify the correct sub-component
System         => _____ System name, sysplex name or domain ID
Scope         => ONLY_____ Change automation for:
                                ONLY - only this resource
                                CHILDREN - dependants
                                ALL - resource and dependants
Automation    => N          Specify new settings for automation flags
Initstart     => -          Y - flag is always on
Start         => -          N - flag is always off
Recovery      => -          E - exits called if specified, otherwise
Shutdown     => -          always on
Restart       => -          R - return to automation control file
                                settings
Interval      => _____ Specify duration of change (dd:hh:mm)
                                or leave blank for an indefinite change

Command ==>
PF1=Help    PF2=End    PF3=Return
PF6=Roll                                PF12=Retrieve
    
```

Figure 38. INGAUTO Command Dialog

In the *Resource name* field, type the name of the resource for which you want to determine automation flags. If you specify the value defaults, you set automation flags for all resources that do not have more specific automation flags defined or that do not belong to a group that has flags defined.

In our example screen from Figure 38, automation is turned off for the TSO resource.

You can use the *Interval* field to turn automation on or off for all or selected flags for a specific time period. In this field, type the amount of time you want the automation turned on or off. This time period begins immediately when you press the Enter key.

At the end of the interval, the flag(s) will be reset to the value defined by your automation control file. If you leave the *Interval* field blank, the flag(s) will remain changed until they are manually reset or updated.

For the complete syntax of the commands introduced in “Determining Automation Agent Flag Values” on page 103, refer to *System Automation for z/OS Operator’s Commands*.

How to Work with Application Groups

An application group is a resource which both summarizes the status of other resources (its members) and which controls the activation of those members. A special feature is that the members can inherit the group’s dependencies.

Each member is a resource, as is the group itself. This means that groups can contain other groups. You can issue commands to start or stop groups as an entity. Also, you can link a group to a service period and thus schedule it automatically.

Detailed information on how to organize applications into a group and how to nest groups into other groups and how to generate resources from groups and applications is provided in *System Automation for z/OS Defining Automation Policy*.

How to Start and Stop Resources

The concept of resources allows you to dynamically move applications and groups from one system to another system in the sysplex. This move is achieved by stopping an active set of resources and starting an alternate set in a coordinated fashion. With this concept, you can also start and stop equivalent resources to maintain a desired level of server capacity.

There are two types of application groups:

- *System Application Groups* - application groups that are associated with particular systems.
- *Sysplex Application Groups* - application groups that are associated with a certain sysplex. The application group members can be spread over multiple systems within this sysplex.

Furthermore, groups can have one of three different natures:

BASIC

In a BASIC application group *all* of its components must be available before the group is considered to be available.

MOVE

In a MOVE application group exactly one of its components must be available before the group is considered to be available. In MOVE groups you can specify alternative components that will be started if the primary component fails to start.

SERVER

In a SERVER application group any number of components can be specified that must be available before the group is considered to be available. In SERVER groups you can specify what should happen if its components fail to start.

The automation programmer can set the numbers of members that should be available for a SERVER group by specifying the *Availability Target* when defining the group.

Preference values are used for SERVER and MOVE groups to determine which members should preferably be made available.

For detailed information on how to define application groups and which nature to use for which purposes, refer to *System Automation for z/OS Defining Automation Policy*.

How to Use INGGROUP to Change Policy for Applications

The functions provided by the INGGROUP command are:

- automatic group management to maintain application availability. This includes the activation of backup/alternate resources in response to resource or system failure.
- a superset of the behavior provided by Automatic Restart Manager. While INGGROUP does not provide a true cross system restart, it is able to take recovery actions in a variety of situations where Automatic Restart Manager is not. These include application failure, supporting application failure, and 'total active instance' based management (for SERVER groups).
- the ability to move resources away from a system, on either a group/subgroup level or a system level.
- the ability to query and dynamically change a group's policy so as to cause one instance of a resource to be stopped and another started.

- the ability to view an estimate of the impact of those changes before committing them.
- the ability to perform actions resulting from the above in a delayed, non-disruptive fashion.
- the ability to return resources to their normal systems in a delayed, non-disruptive fashion.

So one of the most important functions of the INGGROUP command is moving resources around in a sysplex. Moving applications or application groups from one system to another, that means, stopping an active set of applications (dependent and supporting resources) on one system and starting them on another system without loss of information. This can be advisable for the following reasons:

- to ensure availability of resources throughout a sysplex despite single system failure
- to balance workload in a sysplex
- to enable complete applications to be moved away from a certain system if this is needed for other work or maintenance

Application move is implemented in SA z/OS in two flavours:

- a planned or scheduled move, which is defined by automation policy and where the availability of resources is automatically ensured due to defined availability targets and preference values for the single resources. For detailed information, refer to *System Automation for z/OS Defining Automation Policy*.
- an immediate move for emergency reasons; for this purpose, you can use the INGGROUP command which you issue against a group to move it.

When an application has been automatically moved to a backup system after a problem, the policy can be set to return it to its home system either as soon as the home becomes available (causing an application outage) or at the next (scheduled) application outage.

For more information about application move see “Controlling Application Move and Server Management” on page 22.

You can invoke the INGGROUP command dialog either by typing the command name into the command line or by selecting option **J** from the INGLIST command dialog. In both cases you reach the panel shown in Figure 39 on page 110.

How to Start and Stop Resources

```
INGKYGRA                SA z/OS - Command Dialogs
Domain ID = IPUFM      ----- INGGROUP -----      Date = 03/30/04
Operator ID = OPER1    Sysplex = AOCPLEX              Time = 10:29:31

Specify or revise the following data:

System => _____ System name, domain id or sysplex name

Action =>                EXCLUDE-AVOID-INCLUDE or ACTIVATE-PACIFY or
                        RESET, DEFAULT

Group(s) => HUGMGRPAPG/APG_____
                _____
                _____

System(s)=> _____
                _____

Command ==>
PF1=Help   PF2=End   PF3=Return   PF4=Members   PF6=Ro11
                PF10=60                PF12=Retrieve
```

Figure 39. INGGROUP Command Dialog (1)

This panel allows you to change the policy of application groups. You may specify:

- The target sysplex for the command through the system field. If omitted, this defaults to the local system/sysplex. Specify the system name only when viewing/changing the policy of an application group that is not part of the local sysplex.
- The policy change that you want to make through the action field.
- Next you must specify either the name of the target group or a set of templates that will match one or more groups. All policy items work with a list of group name templates and apply to all Move and Server groups matching those names. While applying a policy to a Basic group is not an error, it will not achieve anything either.
- For the Exclude/Avoid/Include option, you can specify a list of systems that the policy is to be set for. Specify a question mark (?) in the **System(s)** line to see the list of systems within the sysplex.

This option indicates that you want to remove movable members from a system (exclude), prohibit SA z/OS from activating movable members on a system (avoid) or that you wish to undo either of the other two actions (include).

Note: The AVOID/EXCLUDE options are only applied to the members of the group which are applications. They are not applied to members of a group which are also groups. In other words, these options are not recursively spread out to nested application groups.

- With the PACIFY option, you can make the specified group passive. This option will immediately revoke all votes from all of the group members. This may cause the desired state of some of its members to change, resulting in them being started or stopped.

With the ACTIVATE option, you can make a group active. When a group becomes active, SA z/OS will propagate *MakeAvailable* requests to the group members that are required to make the group available. Those members which are not required will receive a *MakeUnavailable* request. Note that if a member has a preference of 1 it will always be propagated a *MakeUnavailable* request and if it has a preference of 0 (a passive member) it will never be propagated any sort of request.

- With the RESET option you can set the preference value assigned to each member of the group to its initial value.
- With the DEFAULT option you can set the availability target, satisfactory target, group passive attribute and the member's preference value to the value defined in the policy.

To make your changes active, press PF10. Pressing PF4 from the previous panel will bring you to the one shown in Figure 40 where you can change the preference policy values for the members of a Server group.

```

INGKYGRB          SA z/OS - Command Dialogs          Line 1 of 3
Domain ID = IPUFM  ----- INGGROUP -----         Date = 03/30/04
Operator ID = OPER1       Sysplex = AOCPLEX         Time = 10:30:26

Group: HUGMGRPAPG/APG          Nature: Server          Passive: NO
Description: sample server group
Excluded:
Avoided :
Mode: Normal      Availability Target: 1      Adjust: 0      Result=> 1
                  Satisfactory Target=> 1

Name      Type System  Pref  Adj  Result  Avl  Eff  Stat  Act
-----
HUGMGRP   APL  AOCA    700   0    700   Yes  950  Sel  --
HUGMGRP   APL  AOCC    700   0    700   No   700  Uns  --
HUGMGRP   APL  AOCD    700   0    700   Nst   0   Uns  --

Command ==>
PF1=Help    PF2=End      PF3=Return   PF6=Roll
PF9=Refresh PF10=G0      PF11=Reset   PF12=Retrieve
    
```

Figure 40. INGGROUP Command Dialog (2)

On the panel shown in Figure 40 you can change the values in the **Result** field to change the availability target for that server group. Also, in the **Result** column you can change the preference value for that member. The effective preference value is now shown in the **Eff** column. (Preference value changes are also possible for Server groups).

Changes to preference values for members, as set via the Result field are implemented in the following manner:

1. The preference value in the automation policy is not affected.
2. An adjustment value is calculated, which produces the Result value, when it was applied to the preference value in the automation policy.
3. The Result preference value is used by the automation when a preference value is required.
4. The adjustment value, not the Result, is preserved across automation manager HOT restart and across INGAMS REFRESH, but is lost during WARM and COLD restarts.

To make your changes active, press PF10. To reset the initial settings, press PF11.

The action taken after PF10 is pressed may differ from the assumed or proposed action displayed in the action column of panel INGGROUP. This is because the assumed or proposed action is calculated by processing the group's resources in alphanumeric sequence, while the actual action performed is calculated by the automation manager which processes the group's resources in a random sequence.

How to Start and Stop Resources

For detailed information about the INGGROUP command refer to *System Automation for z/OS Operator's Commands* and to the online help of the INGGROUP command dialog.

How to Set a Timer

You can set a timer to issue commands at a specific time or interval of time for your convenience. Therefore, you do not need to be present at a console to perform a repetitive task at a certain time.

This NetView timer is actually the SETTIMER command that issues a command or list of commands at a specified time or specified interval.

Using the SETTIMER command, displays a list of existing timers as shown in Figure 41. From this panel, you can perform the following tasks:

- “How to Create a Timer”
- “How to Modify a Timer” on page 113
- “How to Delete a Timer” on page 113

```
AOFKATI SA z/OS - Command Dialogs Line 1 of 32
Domain ID = IPSFM ----- SETTIMER ----- Date = 04/15/04
Operator ID = OPER1 System = KEY1 Time = 12:46:40
Enter ADD on the COMMAND line to create a new timer entry
or use CMD selections A add, C change, or D delete to modify existing timers.
CMD TIMER-ID SCHEDULED TIME TYPE TASK INTERVAL CLOCK SAVE CTUP
-----
AOFI0004 04/15/04 12:47:26 AFTER PPT GMT NO
Cmd: AOFRASCD AOFWRK06 AOFRSM00 STARTUP,DB2T,,NORM,NO,,**,YES
AOFI0006 04/15/04 12:47:39 AFTER PPT GMT NO
Cmd: AOFRASCD AOFWRK08 AOFRSM00 STARTUP,CICSK1H,,NORM,NO,,**,YES
AOFI0008 04/15/04 12:47:44 AFTER PPT GMT NO
Cmd: AOFRASCD AOFWRK08 AOFRSM00 STARTUP,OPCD,,NORM,NO,,**,YES
AOFI000B 04/15/04 12:47:50 AFTER PPT GMT NO
Cmd: AOFRASCD AOFWRK02 AOFRSM00 STARTUP,TESTFLAG_I,,NORM,NO,,**,YES
AOFI0005 04/15/04 12:47:59 AFTER PPT GMT NO
Cmd: AOFRASCD AOFWRK04 AOFRSM00 PRESTART,IM631C4,,NORM,NO,,**,YES
AOFI0002 04/15/04 12:53:22 *igno PPT 000 00:30:00 GMT TIME
Cmd: AOFRASCD AOFWRK04 AOFRSMN2 TWSCMSRVVR

Command ==>
PF1=Help PF2=End PF3=Return PF6=Ro11
PF8=Forward PF9=Refresh PF12=Retrieve
```

Figure 41. SETTIMER Command Dialog

How to Create a Timer

Type add on the command line from the panel shown in Figure 41 and press Enter. On the upcoming panel (shown in Figure 42 on page 113), fill in the fields as required.

Note: If you want to add a new timer but use most of the same values as an already-defined timer, type A in the CMD column and modify the SETTIMER panel.

```

AOFKATIR          SA z/OS - Command Dialogs
Domain ID = IPSFM ----- SETTIMER ----- Date = 04/15/04
Operator ID = OPER1          Add/Change Timer      Time = 12:59:18

Timer ID          => EVE00070      on System : KEY2
Type              => EVERY          AT, EVERY, AFTER, CHRON
When?
Interval          =>          days 0-365      and 00:10:00 (elapsed hh:mm:ss)
Specific date     =>          mm/dd/yy      at          (AT or EVERY)
Day of Week       =>          at          (AT or EVERY)

Task              => AUTWRK01      Operator ID or PPT
Save              => NO           No, Yes      Timer clock => GMT   GMT, LOCAL

CHRON parameters =>

Timer command => EVEEMHCN EYUMAS2B 0 EVECHLTH 30 INIT .

Command ==>
PF1=Help  PF2=End  PF3=Return  PF4=Clear Command      PF6=Roll
PF12=Retrieve
    
```

Figure 42. Add/Change Timer

How to Modify a Timer

On the panel shown in Figure 41 on page 112, type C in the CMD column for the timer you want to change. The upcoming panel is the same as shown in Figure 42 where you now can modify the settings of your selected timer.

How to Delete a Timer

On the panel shown in Figure 41 on page 112, type D in the CMD column for the timer you want to delete and press Enter.

How to Interact with the Automation Manager - INGAMS

You can operate the automation manager itself with the INGAMS command in the following way:

- display information about all currently registered automation managers and automation agents.
- change the operation mode of an automation manager from secondary to primary
- refresh the configuration data used by the automation manager

This command displays all currently registered automation managers and automation agents as shown in Figure 43 on page 114. The *Role* describes what you see:

AGENT

denotes an automation agent

PAM denotes a primary automation manager

SAM denotes a secondary automation manager

How to Start and Stop Resources

```
INGKYAM0          SA z/OS - Command Dialogs          Line 1 of 6
Domain ID = IPUFM  ----- INGAMS -----          Date = 06/27/04
Operator ID = OPER1          Sysplex = AOCPLEX          Time = 14:47:44

Cmd:  A Manage      B Show Details  C Refresh Configuration  D Diagnostic

Cmd  System  Member  Role  Status  Sysplex  XCF-Group  Release  Comm
-----
AOCA  AOCA  AOCA  AGENT  READY  AOCPLEX  INGXSGA0  V2R2M0  XCF
AOCA  AOCA  AOCA$$$1  PAM  READY  AOCPLEX  INGXSGA0  V2R2M0  XCF
AOCB  AOCB  AOCB  AGENT  READY  AOCPLEX  INGXSGA0  V2R1M0  XCF
AOCC  AOCC  AOCC  AGENT  READY  AOCPLEX  INGXSGA0  V1R3M0
AOCD  AOCD  AOCD$$$1  SAM  READY  AOCPLEX  INGXSGA0  V2R2M0  XCF
AOCD  AOCD  AOCD  AGENT  READY  AOCPLEX  INGXSGA0  V2R2M0  XCF

Command ==>
PF1=Help      PF2=End          PF3=Return      PF6=Roll
                PF9=Refresh      PF12=Retrieve
```

Figure 43. INGAMS Command Dialog

Entering the following command codes displayed in the upper part of Figure 43, you can initiate all activities concerning automation managers:

- A** You can modify the operation mode of the automation manager from secondary to primary, and enable takeover file processing.
- B** You can have additional information displayed about the automation manager or the automation agents. For an automation agent, the DISPSYS command is issued to show details about the agent and the environment it runs on.
- C** You can have a panel displayed that prompts you for the name of the data set containing the automation manager configuration file members. This panel is used to update the definition data for the automation manager in real time and also for reloading the corresponding automation control files on the active automation agents. This command thus maintains the consistency of the configuration data as used by the automation agent and the associated automation control files.
- D** You can issue this command for a primary automation manager to view or produce diagnostic information, such as recording actions performed during automation. For more information, see “How to Use the Diagnostic Option of INGAMS” on page 277.

How to Change the Automation Agent Status - SETSTATE

You may wish to change the status of an application under the following circumstances:

- When you want SA z/OS to resume handling of an application after a manual operator shutdown or after any other event that causes an application to go to a non-automatable status (for example, STOPPED, CTLDOWN, BROKEN, ACTIVE).
- When you want to prevent automation from restarting an application by setting its state to CTLDOWN.
- When you want to inform SA z/OS that an application that has been shut down is going to be moved to another system.

System operations defines twenty six different statuses for automation, described in "Automation Agent Statuses" on page 38. For details about these statuses, see the help panels or *System Automation for z/OS Customizing and Programming*. See Figure 44 and Figure 45 on page 116 for the relationships of the automation statuses. You can change the states illustrated here with the SETSTATE command (described in the *System Automation for z/OS Operator's Commands*).

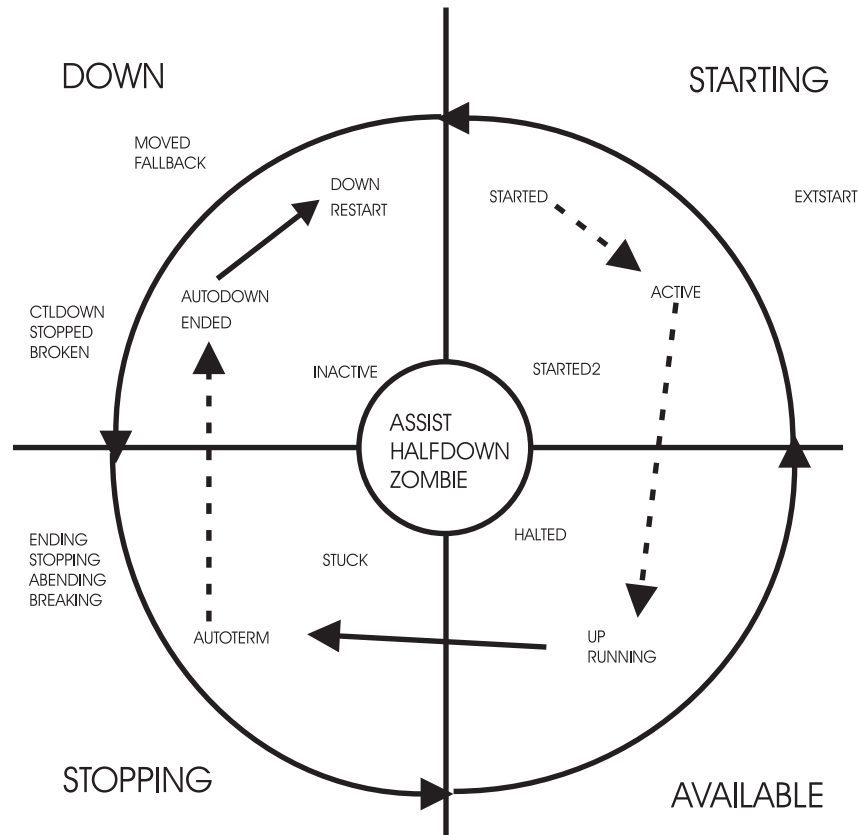


Figure 44. Transitions for a Normal Subsystem

How to Start and Stop Resources

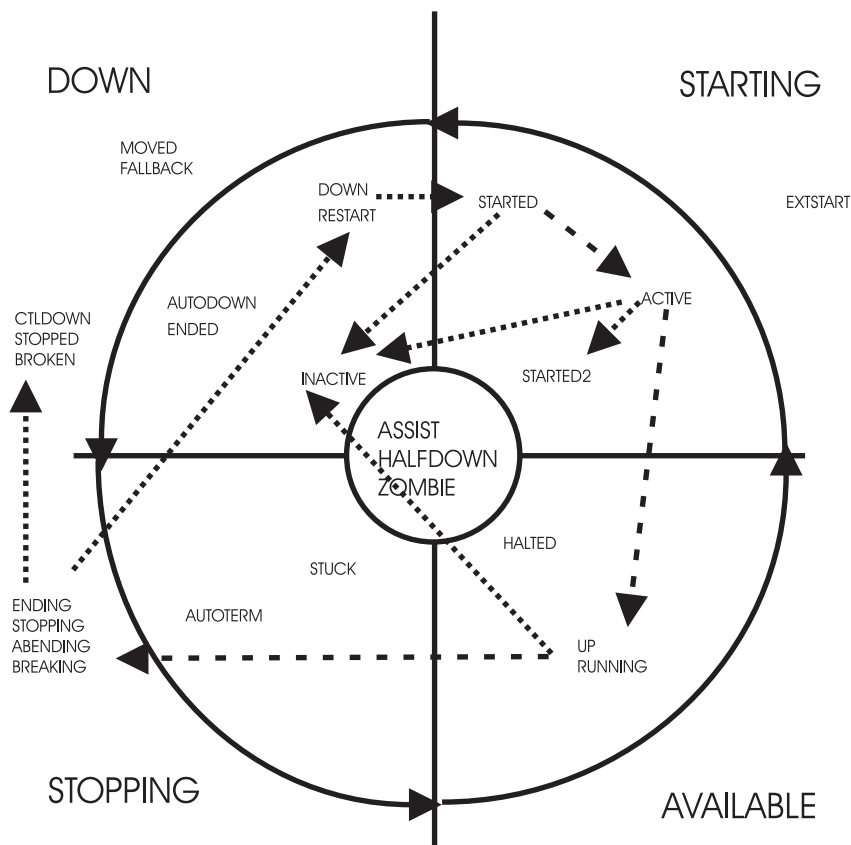


Figure 45. Transitions for an Abending Subsystem

To change the status of a subsystem, issue the SETSTATE command (see Figure 46 on page 117). The SETSTATE command is used to change the automation status of a subsystem or a group of subsystems (supporting and dependent resources). The status change you are able to make depends upon the subsystem's application monitor status. If the subsystem's application monitor status is ACTIVE, you may change its automation status to UP. If its application monitor status is INACTIVE, you can change it to RESTART or CTLDOWN. You may also be able to change it to MOVED or FALLBACK, depending upon its system associations.

With the SETSTATE command dialog, you can change the automation agent status of a subsystem or a group of subsystems by means of the *Scope* field as shown in the panel from Figure 46 on page 117. In this sample panel, the status of the TSO application and all of its dependent applications is set to CTLDOWN.


```

AOFKSAST          SA z/OS - Command Dialogs
Domain ID = IPUFM ----- SETSTATE ----- Date = 03/10/04
Operator ID = OPER1                               Time = 17:24:11

Specify subsystem and new state:

Subsystem => TSO_____ Subsystem to be processed
System    => _____ System name, domain ID, sysplex name or *ALL
State     => CTLDOWN_   Put subsystem in the following state:
                        (RESTART / AUTODOWN / CTLDOWN / UP / ENDED /
                        MOVED)

Scope     => ALL_____ Change state for:
                        ONLY - only this subsystem
                        CHILDREN - only dependant subsystems
                        ALL - resource and dependants
Start     => NO_____ Start or restart subsystem now (YES/NO)
Override  => NO_____ Override(Start=YES) (NO/FLG/TRG/SVP/EVT/ALL)

Command ==>
PF1=Help    PF2=End    PF3=Return
PF6=Roll    PF12=Retrieve
  
```

Figure 46. SETSTATE Command Dialog

Note: The *Override* field is valid only for systems automated with downlevel SA z/OS.

How to Monitor Messages

Message suppression facilities at the target system determine the amount of message traffic sent from a target system sent to a console. Messages that do not normally require operator intervention are usually suppressed at the target system. Messages that do normally require operator intervention may also be suppressed at the target system as automation programming is developed to respond for you.

You can monitor messages at the SA z/OS workstation via several mechanisms and at several levels. You can:

- See no messages
- See a summary of outstanding WTOR messages
- See messages sent to a specific system console or operator console
- See messages related to the control of processor operations-managed resources
- See SA z/OS messages (specified by message class)

How to See Outstanding WTOR Messages

WTOR (Write to Operator with Reply) messages are z/OS messages which require operator action. These messages can be seen on the z/OS operator console.

How to Monitor WTORs on NetView Consoles

Write to Operator with Reply (WTOR) messages are z/OS system messages that inform you of various conditions. WTORs are messages that require a response.

To view a WTOR on NetView operator consoles, use the DISPWTOR command.

For further information on the DISPWTOR command refer to the *System Automation for z/OS Operator's Commands*.

How to Start and Stop Resources

The WTORs are sorted by subsystem, with each subsystem's primary WTOR shown first, and then any secondary WTORs. The primary WTOR is the one that shutdown reply is passed to.

```
AOFKADAC          SA z/OS - Command Dialogs          Line 1 of 5
Domain ID = IPSFM ----- DISPWTOR -----          Date = 04/15/04
Operator ID = OPER1                                     Time = 12:42:33

  Rply System  Subsystem  Message
  ----  -----  -
    95 KEY1    IMS711C4  DFS996I *IMS READY* M711
    88 KEY1    MVSESA   DSI802A IPSNM REPLY WITH VALID NCCF
                                     SYSTEM OPERATOR COMMAND
    93 KEY1    NETATST  DSI802A IPSFM REPLY WITH VALID NCCF
                                     SYSTEM OPERATOR COMMAND

Command ==>
PF1=Help    PF2=End    PF3=Return  PF6=Ro11
             PF9=Refresh PF12=Retrieve
```

How to See Messages for a Specific System or Operator Console

You can choose to receive all messages for a specific target system operator console using the *Processor Operations Interested Operator List*. These messages are displayed at your NetView console session.

Messages from the target console that are displayed include new message lines displayed on the Support Element console integration (CI) window, sent by z/OS, OS/390, VM, Linux for z/Series, VSE, or the Coupling Facility Control Code (CFCC). Multiline messages issued by z/OS look like several single-line messages from the console. Similarly, if an z/OS message extends over two lines, it looks like multiple separate messages from the processor operations console.

When you are monitoring a CI, make sure it is enabled to receive regular messages. For z/OS, refer to the VARY CONSOLE command parameters ACTIVE, DEACTIVATE.

All processor operations messages from a target system begin with the processor operations identifier ISQ900I or ISQ901I plus the target system name and the console type. Messages prefixed by ISQ900I are not displayed but are used by the automation. User automation should key on the ISQ900I messages. Messages prefixed by ISQ901I are sent to interested operators and should not be used for user automation. Multiline messages appear as multiple messages.

Join or Leave an Interested Operator List: For target system operator console messages from CI use the ISQXMON command. For general processor operations resource control information messages use the ISQCMON command on a NetView console.

See the List of Interested Operators: The processor operations status facility contains status panels showing the list of interested operators for a specific target system.

You cannot directly access the *Processor Operations Interested Operator List* panel with the ISQXDST command. You can access the *Processor Operations Interested Operator List* panel only from the following status panels:

- *Target System Summary* panel (Figure 52 on page 127)

- *Target Hardware Summary* panel — accessible only from other status panels

The best way to access the *Processor Operations Interested Operator List* panel is to issue the following command to access the *Target System Summary* panel for a specific target system:

```
ISQXDST target_system_name
```

The resulting status panel shows the communication path status for the connection between the processor operations focal point and the target hardware (CPC) where the target system image is running (LPAR or Basic Mode). Place the cursor on the communication status line and press the PF7 key to see the associated *Processor Operations Interested Operator List* panel. If your operator identifier (the name you use to log on to NetView) is in this list, you should be receiving messages from that path. All other operators whose names are in the list also receive messages from that path.

You check the interested operator list for processor operations resource control messages (the ISQCMON list) by issuing the following NetView command:

```
LIST ASSIGN=GROUP
```

The interested operator list for processor operations is in group +TSCFMON.

How to See SA z/OS Messages Specified by Message Class

Notification operators are defined during customization of SA z/OS, using the SA z/OS customization dialog. A notification operator is a NetView console operator who is authorized to receive messages from SA z/OS. The classes of messages which a notification operator is authorized to receive are specified at the time a notification operator is defined. See the tasks below to understand how to display information about a specific notification operator, how to add and delete notification operators, and how to change the messages which a notification operator receives.

Automation manager messages can automatically be received.

Also, you can receive messages from all systems within the sysplex.

Table 8. SA z/OS Message Classes

Message Class Description	Message Class Number
<i>General Message Classes</i>	
Errors	0
Debug	1
<i>Automation Agent Related Message Classes</i>	
All automation agent messages	40
Down status	41
Up status	42
Information	43
Action	44
Status	45
User intervention	46
Critical threshold setting exceeded	50

How to Start and Stop Resources

Table 8. SA z/OS Message Classes (continued)

Message Class Description	Message Class Number
Frequent threshold setting exceeded	51
Infrequent threshold exceeded	52
Assist mode alert	55
IMS Automation Messages	60
<i>Automation Manager Related Message Classes</i>	
All automation manager messages	80
Intervention required	81
Automation denied	82
Automation Impacted	83
Awaiting Automation	84
In Automation	85
Degraded - Target resources	86
Degraded - All resources	87
Satisfactory - Target resources	88
Satisfactory - All resources	89
SA z/OS Infrastructure/Environment	90

How to Get Notify Messages - INGNTFY

This section gives you all the information you need about notification operators and those messages that notification operators should be notified of.

Display Settings for a Notification Operator: If no parameter is specified in the INGNTFY command, the following panel is displayed. The panel can be used to:

- Add *A* notify operator using the settings of the selected operator
- Show/Change settings *C* for selected operator
- Delete *D* notify operator
- Turn off *O* notify operator message receiving

```

AOFKAANT                SA z/OS - Command Dialogs          Line 1 of 3
Domain ID = IPUN8      ----- INGNTFY -----          Date = 12/20/04
Operator ID = OPER1                                       Time = 13:52:17

Cmd:  A Add      C Show/Change settings  D Delete  O Turn OFF msg receiving

Cmd Operator System  Log Rcv Description
-----
- OPER1   AOC8    Y  Y  USER1
- USER2  AOC8    Y  N  USER2
- USER3  AOC8    Y  Y  USER3

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Ro11
               PF9=Refresh  PF12=Retrieve
    
```

For further information on the INGNTFY command refer to the *System Automation for z/OS Operator's Commands*.

Query a Specific Notification Operator: To display information about a specific notification operator, type INGNTFY on any command line and type query into the *Status/Action* field. Also, type the ID of the operator you want information about (in the example below, the operator ID is "NETOP1").

```

AOFKAASN          SA z/OS - Command Dialogs
Domain ID = IPSFM ----- INGNTFY -----      Date = 04/15/04
Operator ID = OPER1                               Time = 12:49:43

Specify operator authority:
  Status/Action =>                               Operator status:
                                                    ON - Set your notifications ON
                                                    OFF - Set your notifications OFF
                                                    ADD - Add or Modify an operator
                                                    DELETE - Remove an operator
                                                    QUERY - Look up an operator
Operator ID =>                                     Operator for ADD, DELETE or QUERY
Classes =>                                         Up to 10
Description =>

Held Messages - Information => Immediate Action =>
                  Eventual Action => Immediate Decision =>
                  System Wait =>

System =>                                           System name, domain ID or sysplex name

Command ==>
  PF1=Help    PF2=End    PF3=Return                                PF6=Ro11
                                                    PF12=Retrieve
  
```

Turn On or Off all Notification Messages: Use INGNTFY OFF on a NetView or operator console to temporarily turn off messages when you do not want to see them. Use INGNTFY ON on a NetView or operator console to turn the messages back on when you wish to see them again.

Add a Notification Operator: You can add more notification operators through the INGNTFY command dialog. Remember that this operator will only be able to receive messages until the next time the automation control file is loaded, when all changes made using the operator interface are lost.

To add a notification operator, type INGNTFY on any command line. You see the panel as shown on page 121. Type add into the *Status/Action* field. Also, type the ID of the operator you want to add.

In the Classes field, type the class(es) of messages you want this operator to see. (See Table 8 on page 119 for a list of message classes.) The default classes are 40 and 80.

If you want to hold certain types of messages on this operator's screen, move the cursor beside the desired Message type (Information, Eventual Action, Immediate Decision, System Wait, or Immediate Action), and type any character. For more information on message types, see *System Automation for z/OS Messages and Codes*.

Change the Messages a Notification Operator Receives: To change the messages you or another operator receives, you access the INGNTFY command dialog and you fill in the panel exactly as if you were adding a new operator. (You do not have to delete the definition first because the new definition overwrites the old.)

Because the new definition overwrites the old, you need to be careful when you are changing the messages an operator receives. For example, if the operator was originally assigned to receive classes 41 and 43, and you typed 44 in the Classes

How to Start and Stop Resources

field, 44 would be the only class that the operator would now receive. If you want to add a class, you must also type all the classes that were previously assigned.

You can perform a query to find out what messages the operator currently has assigned. See task "Query a specific notification operator" (page 121) for more information.

Delete a Notification Operator: You can also use the INGNTFY command dialog to delete notification operators. Remember that if you use this panel to delete an operator, the deletion will be in effect only until the automation control file is reloaded.

To delete a notification operator, type delete into the *Status/Action* field. Also, type the ID of the operator you want to delete. This field can contain any valid NetView operator or the keyword SYSOP. If SYSOP is defined as a notification operator its function is to receive messages to the master console when no other operator is logged on, or to receive specific messages depending on the message classes assigned.

How to Monitor Hardware

You can monitor all hardware, from processors to devices, using processor operations and I/O operations.

You can also monitor hardware resources and you can issue system operations and processor operations commands from the NetView Management Console as described in Chapter 11, "Using the NetView Management Console for SA z/OS," on page 217.

Some General Information About Panels: Pressing the Enter key toggles between a dynamically updated panel and a static panel. If any processor operations screen is updated frequently, it appears to flicker. You may need to place the panel into a static condition to select another function. The Updates field in the top right of the panel shows the current mode of the panel (Dynamic or Static).

Processor Operations Status Information

Processor operations allows you to monitor target processors, target systems, and focal point communication path by using the ISQXDST status panels of your NetView operator console. These panels are available only on a NetView operator console attached to the SA z/OS processor operations focal point system.

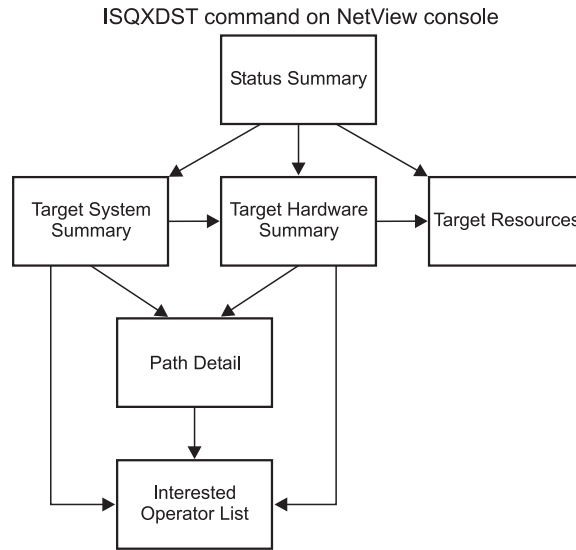


Figure 47. ISQXDST Command on NetView Console

Displaying Processor Resource Information

To view target hardware summary of a processor, select PF9 from the *Target System Summary* panel (Figure 52 on page 127) or from the *Status Summary* panel (Figure 51 on page 126) on a NetView console.

```

ISQEHARP          SA z/OS Proc-Ops THW Summary  Updates: Dynamic
Target Hardware Name: YORAMA  Type: 9672    Mode: LPAR
                          Model: RX4      Serial Number: 51-0064523
Target Hardware Description: Web Services Backbone Bldg. 14
Initialized Target Systems:
    KEY6      CF61      CF62      KEY7
Communication Path Error Status:
Last Significant Message:

Enter=Static PF1=Help  PF3=Exit  PF6=Roll  PF7=Oper List  PF11=PATH Dtls  PF12=Quit
  
```

Figure 48. Target Hardware Summary Panel for OCF-Based Processors

This panel defines the PF keys as follows:

- The PF7 key displays the *Processor Operations Interested Operator List* panel (the one accessed with the ISQXMON command).
- PF11 displays the *Path Detail* panel (Figure 53 on page 131). This panel provides detailed status information about a specific NetView path. First place the cursor on the line with the name of the path for which you want more information, and then press PF11.

The *Target Hardware Summary* panels show the following specific information:

How to Start and Stop Resources

Target Hardware Name

The name assigned to this definition of target hardware in the customization dialog.

Target Hardware Type

Indicates the machine type of a zSeries or 390-CMOS processor.

Target Hardware Mode

LPAR, ESA

Target Hardware Description

Short textual description of this definition of target hardware, defined in the customization dialog.

Initialized Target Systems

(updated dynamically) List of target systems currently initialized to this target hardware. For non-LPAR-mode target hardware, this field lists one or none. For LPAR-mode target hardware, this field can list multiple target systems. This field is blank if no target systems are initialized to this target hardware.

Communication Path Error Status

(updated dynamically) Status information in case of a communication problem between the NetView focal point and the target hardware.

Valid values for the status of the NetView connection path are:

NVC or SNMP SESSION BUSY

(red) Processor operations cannot presently communicate with the target support element, the problem appears recoverable.

NVC or SNMP SESSION PROBLEM

(red) Processor operations cannot communicate with the target support element.

Last Significant Message

The last significant message or alert associated with this target hardware

How to Monitor Health Performance

You can monitor the health performance of your resources using monitor resources with the NCCF panels or the NMC.

Using the NCCF Panels

Once you have loaded your configuration you can use the NCCF panels to check the status of your resources. You might want to start with INGLIST. If you scroll to the right, you will see the new health status field.

Now assume that you are monitoring CLIENT/APL with INGLIST and you find a compound status of DEGRADED and a health status of WARNING. In this case you can enter the line command M in front of CLIENT/APL. This will take you to the DISPMTR panel showing all MTRs that are connected to CLIENT/APL via HasMonitor relationships. In this case you would see a panel similar to that shown in Figure 49 on page 125.


```

INGKYMO0          SA z/OS - Command Dialogs      Line 1    of 1
Domain ID   = IPUNG          ----- DISPMTR -----      Date = 06/17/04
Operator ID = USER1          System = AOC1              Time = 11:05:52

CMD: A Reset   B Start   C Stop   D Details   F INGINFO   I INGSCHED

CMD Monitor      System      Status      Health      Last monitored
-----
_  MON1          AOC1        ACTIVE      WARNING     2003-12-17 11:05:35

Command ==>
PF1=Help   PF2=End       PF3=Return          PF6=Ro11
           PF9=Refresh PF10=Previous PF11=Next           PF12=Retrieve
    
```

Figure 49. DISPMTR Command Dialog Panel

In order to find the reason for the health status WARNING you can scroll to the right to find more information or enter the line command D in front of MON1/MTR which takes you to the DISPMTR Details panel, as shown in Figure 50.

```

INGKYMO1          SA z/OS - Command Dialogs      Line 1    of 22
Domain ID   = IPUNG          ----- DISPMTR -----      Date = 06/17/04
Operator ID = USER1          System = AOC1              Time = 11:10:09

Monitor       : MON1/MTR/AOC1
System        : AOC1
Description   : passive monitor

Commands...
  Activate    :
  Deactivate  :
  Monitoring  : MYMON 9.152.142.72

Interval      : 00:10

Monitor Status : ACTIVE at 2004-06-17 11:08:36
Health Status  : WARNING
                PING lasted 15 ms

History...
                2004-06-17 10:38:34 - ACTIVE      HEALTH=NORMAL
                PING lasted 1 ms
                2004-06-17 10:48:34 - ACTIVE      HEALTH=NORMAL
                PING lasted 1 ms
                2004-06-17 10:58:35 - ACTIVE      HEALTH=NORMAL
                PING lasted 1 ms

Command ==>
PF1=Help   PF2=End       PF3=Return          PF6=Ro11
           PF9=Refresh PF10=Previous PF11=Next           PF12=Retrieve
    
```

Figure 50. DISPMTR Command Dialog Details Panel

Using the NMC

To use the MTRs in your NMC you can define a view that will contain one or more MTRs by entering the appropriate filter criteria. If you want to see your APL together with the connected MTRs you must work out a naming convention that allows you to specify wildcards. Another (less flexible) possibility is to name every resource (APL and MTR) explicitly in the view.

How to Start and Stop Resources

If you have a view with both APLs and connected MTRs you will see the HasMonitor relationships as lines between them. Now you can right-click on the icons and issue the associated commands such as DISPMTR or INGMON.

How to Monitor Processor Status

Processor status can be monitored by viewing the status of a system icon on the NMC workstation or processor status can be seen in more detail using the processor operations 3270-type panels. On the NMC workstation, the status of a system icon represents the aggregation of status of the target's operating system software and its underlying processor hardware. You can see at a glance whether the systems for which you are responsible have satisfactory status.

Refer to Chapter 11, "Using the NetView Management Console for SA z/OS," on page 217 for information on how to enter commands against an unsatisfactory processor operations object.

This section contains the following subsections that describe certain aspects of monitoring processor status:

- "View Status Summary of all Target Systems"
- "View One Specific Target System" on page 127

View Status Summary of all Target Systems

On a 3270 *console*, issue `isqxdst` (with no parameters) on a NetView console. The *Status Summary* panel is displayed.

```
ISQESUM SA z/OS Proc-Ops Target Status Summary      Updates: Dynamic 1
Control file in use: SATEST.SYSTEM.V22.IPSFO.ISQCNTL NMC Bridge: INACTIVE
I isqxiii C isqxcls O isqxopt  A,B view netlogs E events Debug Mode: OFF

Cmd   Target System  Status 3 Focal Points - Primary: IPSFO Backup: IPUFM

CFA           CLOSED
CFB           CLOSED
CFF           CLOSED
CF1           UNKNOWN
CF61          INITIALIZED
CF62          INITIALIZED
KEYA          STAGE-1 ACTIVATE COMPLETE
KEYB          IPL COMPLETE
KEYC          IPL COMPLETE
LNX0L11      NOT ACTIVE
TEL01FE      LOAD FAILED

Enter=Static PF1=Help PF3=Exit PF4=Tgt Sys Summary PF5=Debug On/Off 2
PF6=Roll    PF7=Up    PF8=Down PF9=Tgt HW Summary PF11=PATH Details
```

Figure 51. ProcOps Target Status Summary Panel

- 1 This indicates whether updates are static or dynamic (it changes automatically whenever the status changes). You can toggle to a static state with the Enter key (for example, when you want to enter a choice).
- 2 The PF keys allow you to invoke other status panels.
- 3 The status that you see for a target system represents the most severe condition that exists for that system. These values are extracted from the *status*, *target hardware*, *attention*, and *console summary* status fields of the *Target System Summary* panel (see illustration in task below).

View One Specific Target System

On a 3270 console, use ISQXDST followed by the name of the target system on a NetView console. The *Target System Summary* panel is displayed.

```

ISQETARG          SA z/OS Proc-Ops Target System Summary   Updates: Dynamic

Target System
Name              : KEY3
Description       :
O.S.              : MVS
Load Profile      :
Status            : UNKNOWN
Attention         :

Target Hardware
Name              : FREEWAY      Mode : LPAR
Attention        :
Path Status      : DORMANT

Target Hardware LPAR
Name              : KEY3          Mode : ESA
Image Profile    : KEY3

Last Significant Message:

Enter=Static  PF1=Help  PF3=Exit  PF6=Roll  PF7=Oper List
PF9=Target Hardware  PF11=Path Detail  PF12=Quit

```

Figure 52. Target System Summary Panel

This panel defines the PF keys as follows:

- The PF7 key displays the *Processor Operations Interested Operator List* panel (the one accessed with the ISQXMON command).
- The PF9 key displays the *Target Hardware Summary* panel (Figure 48 on page 123). This panel provides detailed status information about the target hardware on which the target system is defined.
- The PF11 key displays the connection *Path Detail* panel. This panel provides detailed status information about a specific connection path.

The *Target System Summary* panel shows the following specific information:

Target System Name

The name assigned in the customization dialog to this target system.

Target System Description

Short textual description of this target system, defined in the customization dialog.

Status (updated dynamically) The current value of the processor operations internal variable *tstat*.

You can change this status from INITIALIZED, LOAD FAILED, or IPL FAILED to IPL COMPLETE by issuing the ISQVARS command to change the internal variable *tstat*. (For more information about specifying *tstat* as a keyword on the ISQVARS command, see the description of the ISQVARS command in *System Automation for z/OS Operator's Commands*). You would want to do this when you perform a cold start of processor operations while a target system is already running. After the cold start, you issue the ISQXIII command to the target system that is already running so its status becomes INITIALIZED, then you change the value of *tstat* to IPL COMPLETE. You may also want to change the status manually

How to Start and Stop Resources

if the cause of the LOAD FAILED status was corrected by using the pass-through facility of processor operations, or if it was corrected locally at the site of the target system.

You can also issue the ISQVARS command to change the internal variable *tstat* from IPL COMPLETE to IPL FAILED. Processor operations sets a status of IPL COMPLETE when it initiates a load of a target system and receives an operating system specific message which indicates, the operating system received a level where it is ready to work. However, neither of these operating systems provide a sufficient indication that the load process failed, preventing processor operations from changing the target system status appropriately.

Valid values for this status field are as follows:

CLOSED

(green) The target system has been closed.

INITIALIZED

(green) The target system has been successfully initialized.

IPL COMPLETE

(green, yellow, or red) For a TPF or VSE system, the Load or Load Clear process started. For an z/OS system, processor operations automation responded to the IEA347A message. For a VM system, processor operations automation responded to the VM start message. If this condition is displayed in red, this is done to get your attention to inform you that something is wrong (such as processor operations lost its communication link with the target system) rather than that the IPL process failed. This condition will revert to being displayed in green when the problem is corrected.

IPL FAILED

(red) z/OS only — Error messages were generated during the initial program load process.

LOAD FAILED

(red) The Load or Load Clear operation did not complete successfully.

NOT ACTIVE

(yellow) Activate response was received from the target and did not indicate that stage-1 activation was complete. If the target is running in an LPAR, the logical partition is found not activated.

NOT OPERATING

(red) Target hardware indicates a not operating condition for the image. No CP status is available for problem determination.

STAGE-1 ACTIVATE COMPLETE

(yellow) The Support Element power-on reset has occurred but the load is not yet complete.

UNDECIDABLE

(yellow) SNMP Path Only: The target hardware is not communicating with the support element or the support element is not communicating with the HMC.

UNKNOWN

(green) No attempt has been made to initialize or activate this target system. This status can be set only by a cold start of processor operations.

WAITING FOR IEA101A START MESSAGE

(yellow or red) z/OS only: ACTIVATE, ISQXIPL, or ISQXLOD command processing has completed for an z/OS system and processor operations is waiting for the IEA101A message on the operator console. If this condition is displayed in red, it is to get your attention to inform you that something is wrong (such as processor operations lost its communication link with the target system) rather than that processor operations has been waiting for the message too long.

WAITING FOR IEA347A MESSAGE

(yellow or red) z/OS only: processor operations has replied to the IEA101A message and is waiting for the IEA347A message on the operator console. If this condition is displayed in red, this is done to get your attention to inform you that something is wrong (such as processor operations lost its communication link with the target system) rather than that processor operations has been waiting for the message too long.

WAITING FOR VM START MESSAGE

(yellow or red) VM only: processor operations is waiting for the first VM IPL message on the Support Element CI. The ACTIVATE or LOAD command processing has completed for a VM system and processor operations is waiting for the first VM IPL message. If this condition is displayed in red, it is to get your attention to inform you that something is wrong (such as processor operations lost its communication link with the target system) rather than that processor operations has been waiting for the message too long.

Target Hardware

(updated dynamically) The current value of the processor operations internal variable *thwstat*. Valid values are as follows:

blank (not highlighted) No problem has been reported for the target hardware. You can also use the ISQVARS command to change *thwstat* to blanks after correcting an error condition reflected in this field.

DEGRADED

(yellow) SNMP Path Only: Target hardware runs with degraded performance (specific hardware types only).

POWERED OFF

(red) The CPC is physically powered off.

POWERSAVE

(red) SNMP Path Only: Power utility for the target hardware failed. Target hardware put in power save state.

SERVICE

(red) SNMP Path Only: Service status enabled for the target hardware.

SERVICE REQUIRED

(yellow) SNMP Path Only: The next disruption in the target hardware will result in degraded capacity or will fail.

TARGET HARDWARE PROBLEM

(red) The Support Element of the processor (CPC) reported a target hardware problem.

How to Start and Stop Resources

Attention

(updated dynamically) The current value of the processor operations internal variable *tattn*. Valid values are as follows:

blank (not highlighted) The target system has not encountered any of the following conditions. You can also use the ISQVARS command to change *tattn* to blanks after correcting an error condition reflected in this field.

ALERT

(yellow) SNMP Path Only: Target system attention status indicating an alert has been detected.

DCCF (red) z/OS only — Processor operations has detected a DCCF message.

DISABLED WAIT

(red) Processor operations has detected a Disabled Wait condition.

ENV ALERT

(yellow) SNMP Path Only: Target system attention status indicating an environment alert has been detected.

SERIOUS ALERT

(red) SNMP Path Only: Target system attention status indicating a serious alert has been detected.

Target Hardware

The name of the target hardware on which this target system is running.

O.S. MVS, VM, VSE, LINUX, or CF

Mode LPAR, ESA, 370

LPAR name

The name of the logical partition in which this target system is running.

Connection Path Information

Identifies the connection type and current status of the connection between the processor operations focal point and the CPC support element or HMC.

Type NVC (SNA based NetView connection), SNMP (IP based Simple Network Management Protocol connection).

Status (updated dynamically) The current value of the processor operations internal variable *scstat*.

blank (not highlighted) Either the target system has not encountered any of the following conditions, or the condition has been resolved.

PATH PROBLEM

(red) The focal point system cannot communicate with the Support Element on the NetView connection. Look at the Path Detail panel for more information (Figure 53 on page 131).

ACTIVE

(green) The path is ready for communications.

NVC SESSION BUSY

(red) Processor operations cannot presently communicate with the target Support Element, the problem appears recoverable.

NVC SESSION PROBLEM

(red) Processor operations cannot communicate with the target Support Element.

SNMP SESSION BUSY

(red) Processor operations cannot presently communicate with the target Support Element, the problem appears recoverable.

SNMP SESSION PROBLEM

(red) Processor operations cannot communicate with the target Support Element.

Lock Holder

(updated dynamically) The current NetView operator owning the lock for the target system and the connection path. Locks can be set and unset using the ISQXLOC and ISQXUNL commands. Locks grant the exclusive use of a connection path of a target hardware and its associated target systems.

Last Significant Message

(updated dynamically) The text of the last command response or report received from the target system. The date and time of the message are also displayed. The text of the message is truncated to 79 characters, including the message identifier but not including the time stamp. The message is not removed from the panel when the condition that generated the message is resolved.

How to Monitor Communications Path Status

The communications paths from the processor operations focal point system to target systems are of two types: NVC and SNMP. For these types of path, you can display details of the connection.

Display Path Details: Press PF11 at the *Target System Summary* panel or at the *Target Hardware Summary* panel.

ISQEPNVC	SA z/OS PATH Detail	Updates: Dynamic	
Target Hardware Name: FREEWAY		Connection Type: SNMP	
Network ID	: DEIBMD1	LU Name	: X7F1E30A
Support Element IP Address or Hostname: 9.164.172.200			
Hardware Management Console IP Address or Hostname: SAHMC			
Connection Authorization: POKLICKO		Lock Holder	: IGORDON
PATH Status : ACTIVE			
Poll Frequency : 30			
Poll Retries : 2			
Last Significant Message:			
Enter=Static	PF1=Help	PF3=Exit	PF6=Roll
PF7=Oper List	PF12=Quit		

Figure 53. Path Detail Panel

The *Path Detail* panel shows the following information:

How to Start and Stop Resources

Target Hardware Name

The name assigned in the customization dialog for the processor to which this connection path details belong.

Connection Type

Type of path assigned in the customization dialog to be used for communication between the processor operations focal point and the target hardware. Connection types can be the following:

SNMP

IP based communications protocol which requires that the address information for either SE or HMC are defined. If both addresses are customized in the customization dialog, the SE address has priority.

NVC SNA based communications protocol requiring the CPC SNA address.

LU Name

The NAU part of the CPC SNA address, which is also the logical unit name assigned to the NetView connection path.

Network ID

The NETID part of the CPC SNA address, which is also the name of the SNA network ID for the NetView connection.

Support Element IP Address or Hostname

The name or IP address assigned in the customization dialog to the SE of this target hardware.

Hardware Management Console IP Address or Hostname

The name or IP address assigned in the customization dialog to the HMC of this target hardware.

Connection Authorization

In case of an NVC connection, the authorization token is displayed. For SNMP connections the community value is shown. The authorization values are defined in the customization dialog for a selected target hardware.

Lock Holder

(updated dynamically) The current NetView operator owning the lock for the connection path. Locks can be set and unset using the ISQXLOC and ISQXUNL command. Locks grant the exclusive use of a connection path.

Path Status

(updated dynamically) The status of the NetView path. Valid values are:

UNKNOWN

(green) No command was ever issued to this path name.

ACTIVE

(green) The path is ready for communications.

PATH PROBLEM

(red) The focal point system cannot communicate with the Support Element on the NetView connection. This status is accompanied by a Last Significant Message.

Path Frequency

The time (in minutes) that processor operations should poll the Support Element for status.

Poll Retries

(updated dynamically) The number of attempts to retry that should occur before processor operations notifies the operator that the connection failed.

Last Significant Message

(updated dynamically) The text of the last command response or report received from the target system. The text of the message is truncated to 79 characters, including the message identifier but not including the time stamp. The message is not removed from the panel when the condition that generated the message is resolved.

How to Monitor Alerts on a NetView Console

Alerts are high priority events that require operator action.

Issue NPDA from the command line of your NetView operator's console. The NPDA (Network Problem Determination Application) menu will be displayed. You can request a dynamic display of alerts. From that display you can select an individual alert and see event detail and recommended action. For more details on the use of the NPDA panels, refer to *Tivoli NetView for z/OS User's Guide*.

Monitoring with the Status Display Facility (SDF)

The Status Display Facility (SDF) displays the status of various resources of an z/OS system, and of other z/OS systems that specify this system as an automation focal point. The resources that can be displayed by SDF include:

- applications and subsystems
- WTORs
- gateways.
- application groups

Note: A *gateway* is a combination of a NetView-NetView task session and two automation operators (one on each of two systems) that allows communication of messages, commands, and responses between the two systems.

SDF also shows spool usage problems and ASSIST requests from z/OS subcomponents. SDF consists of a hierarchy of dynamically updated panels showing color-coded status conditions. SDF is set up during the customization of SA z/OS.

Use the status display facility (SDF) to monitor the status of application software, WTORs and gateways on your systems from a NetView or operator console.

All the resources you need to monitor are set up by hierarchical arrangement during customization. At this time, the system programmer:

1. Defines the SDF hierarchy.

The SDF hierarchy indicates which automated resources are affected by status changes in other automated resources (see "How SDF Works" on page 137). The SDF hierarchy can be different from that specified in the automation control file for system startup and shutdown. Use whatever makes sense for your system.

If SDF displays more than one system, programmers create an SDF hierarchy for each system.

Monitoring with the Status Display Facility (SDF)

2. Creates the panels that show the systems and subsystems. These panels can also be customized to display gateway sessions and WTORs.
3. Determines the status colors and other screen defaults for the enterprise or takes the defaults for these attributes.

For information on setting up SDF, see *System Automation for z/OS Programmer's Reference* and *System Automation for z/OS User's Guide*. Because SDF can be customized, your panels may not look like those shown in this topic.

SDF Focal Point Monitoring

At first you need to construct an SDF screen that contains all systems within your sysplex. This will give you an *at a glance* panel that will be dynamically updated with the overall status of each system in your sysplex. By selecting an individual system, you can view a more detailed display of the state of the resources on that system.

On the focal point, you can now issue a command, specifying a target system. The command interacts with the target system (via XCF or RMTCMD) and gives you a full screen display as if you were logged on to the target system.

This means that from SDF you can issue DISPINFO on a target subsystem by simply pressing a key. This allows you to see detailed information about a subsystem that has a problem.

Other commands that support this kind of usage include:

- DISPERRS
- DISPFLGS
- DISPSTAT
- DISPTREE
- INGINFO
- INGREQ
- INGTHRES
- INGVOTE
- SETSTATE

All of these commands can be used from the focal point to interrogate the target system without having to log on to it.

Note

If the target system is not in the same sysplex as your focal point system, then RMTCMD will be used for the inter system communication. This means that there is a dependency upon VTAM being active on both the focal point and the target system for communication to be possible.

If the target system is in the same sysplex, then communications occur via XCF and you do not need to have VTAM active on the target system to talk to it.

Interpreting Colors in the SDF and DISPSTAT

In SDF and on the DISPSTAT command dialog, subsystems appear in different colors which indicate their status. In SDF, the condition of WTORs and gateways is also indicated by color.

Subsystem Colors and Priorities

The color of a subsystem on a DISPSTAT command dialog panel and the color and the highlighting of a subsystem in SDF indicate the current status of a subsystem (the possible status conditions are defined in “Automation Agent Statuses” on page 38). The following list shows the default color (and highlighting) assigned to each status when SA z/OS was shipped. White is also used by default to identify status components without a tree structure. These colors may have been changed for your SDF display. Blinking as a means of highlighting is not used by all types of display, in particular when you are using 3270 terminal emulation on a PC. You may see another type of highlighting, for example, black text on a white background.

Table 9. Table of Default Subsystem Status Colors and Priorities

Color	Highlight	Status	Priority
Red	Reverse	BROKEN	120
Red	Blinking	BREAKING	130
Red	Underline	INACTIVE	140
Red	Normal	STOPPED	150
Pink	Normal	HALFDOWN	220
Pink	Blinking	STARTED2	230
Pink	Underline	STUCK	240
Pink	Reverse	ZOMBIE	250
White	Reverse	ABENDING	320
White	Underline	HALTED	330
White	Blinking	ASSIST	340
Yellow	Reverse	STOPPING	420
Yellow	Normal	AUTOTERM	430
Yellow	Underline	ENDING	440
Turquoise	Blinking	RUNNING	520
Turquoise	Reverse	ACTIVE	530
Turquoise	Underline	STARTED	540
Turquoise	Normal	RESTART	550
Turquoise	Blinking	EXTSTART	550
Green	Normal	UP	640
Green	Underline	ENDED	650
Blue	Underline	DOWN	730
Blue	Normal	AUTODOWN	740
Blue	Blinking	CTLDOWN	750
Blue	Reverse	MOVED	760
Blue	Normal	FALLBACK	770

Monitoring with the Status Display Facility (SDF)

Each automation status is assigned a priority number, which SDF uses to determine what color to display if there are multiple status conditions present in a system. The status conditions are listed in the preceding table in order of descending priority. In other words, the first status listed is the status with the highest priority, and the one whose color will dominate. All of the colors and priorities listed above are defaults, and can be modified when a customer defines SA z/OS policy.

Note: Panels can be modified according to your needs.

MVS Subcomponent Colors

Blinking white is used to display MVS subcomponents that are in ASSIST mode.

Gateway Colors

Besides displaying the status of subsystems, SDF uses colors to show the status of gateway sessions. The following list explains what the gateway colors indicate:

Color	What it means
Reverse Turquoise	Gateway is active
Red	Gateway is inactive

Spool Colors

SDF uses the following colors to show the status of spools.

Color	What it means
Green	Spool is OK
Yellow	Spool is short
Red	Spool is full

WTOR Colors

SDF also uses colors to show what type of WTOR each displayed WTOR is, based on classifications that have been specified using the customization dialog. See *System Automation for z/OS Defining Automation Policy* for more details.

Color	What it means
Green	WTOR is normal
Yellow	WTOR is unusual
Red	WTOR is important

Using SDF

You use SDF to monitor the status of subsystems, WTORs, and gateways on your system and other systems. To monitor status, you watch the highest level SDF panel (also known as the root panel or system summary panel). The following example shows a panel with the systems of a sysplex. This panel displays for example, systems, subsystems, WTORs or gateways within this sysplex.

Monitoring with the Status Display Facility (SDF)

SYSTEM		SA z/OS - SUPPORT SYSTEMS			
System	Subsystems	WTORs	Gateways	Products	System
KEY1	HUGC2APL0	NETATST1	IPSFPO	C I D O	P V M B T U
KEY2	IMS723C4	IMS721M1	IPSFNO	C I D O	P V M B T U
KEY3	IMS733C4	NETATST3	IPSF00	C I D O	P V M B T U
KEY4				C I D O	P V M B T U
XXXX				C I D O	P V M B T U

05/12/04 08:45

====>

1=HELP 2=DETAIL 3=RETURN 6=ROLL 8=NEXT SCR 10=LEFT 11=RIGHT 12=TOP

Figure 54. Example SDF System Summary Panel

On this panel, each system is displayed in a color that reflects the highest priority status of the resources in that system. If a resource in a system changes status, the system changes color to reflect the new status. By watching this panel you will see status changes in any system.

Imagine that **KEY1 HUGC2APL0** is displayed in red reverse on your screen, which means that the subsystem HUGC2APL0 on system KEY1 is in status BROKEN. To view more information on this subsystem, move the cursor to HUGC2APL0 (or KEY1) and press PF2 to get a panel as shown in “How to Check Resource Status Conditions” on page 138).

How SDF Works

SDF uses the hierarchy established during SDF customization to propagate colors up to the highest level panel. The following figure shows a graphic representation of an example SDF hierarchy for KEY1:

Monitoring with the Status Display Facility (SDF)

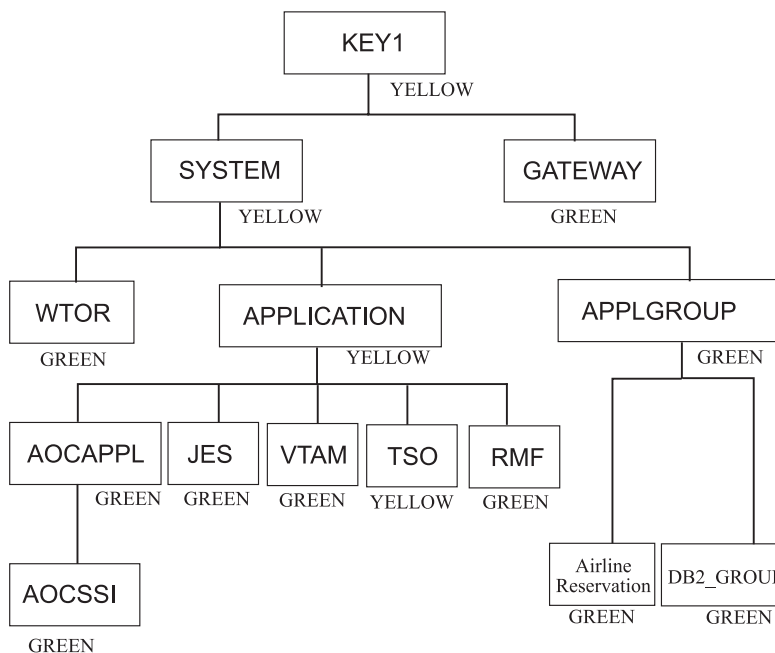


Figure 55. Example SDF Hierarchy for KEY1

Imagine that TSO appears in yellow on your screen because it is in AUTOTERM status. This color is reflected up the hierarchy to APPLICATION, SYSTEM, and KEY1, because AUTOTERM has a higher priority than the status UP (indicated by the color green). If various resources in a system have different status conditions, the one with the highest priority status controls the color of that system. This allows you to resolve issues concerning the highest priority status first, then deal with the lower priority status conditions.

How to Access SDF

To access SDF, type `sdf` on a NetView command line or type 3 (SDF) on the command line of the SA z/OS System Operations menu (see “Using Command Dialogs” on page 68).

You see the highest level (or root) panel of SDF. Figure 54 on page 137 shows an example of the highest level panel for the SA z/OS supported systems. On this panel, **KEY1 HUGC2APL0** is highlighted to indicate that it appears in red on your screen, which means that subsystem HUGC2APL0 on KEY1 is BROKEN. All other systems may appear in green, which means UP.

How to Check Resource Status Conditions

Normally when you use SDF to monitor resources, you watch the highest level panel until you see a system change color. This color change indicates a change in status in one of the resources in the system. To determine which resource, and to see more information on the status change:

1. Move the cursor to the system (or subsystem) that shows a status change.
2. Press PF2. This takes you to the Detail Status Display panel for the resource that caused the system to change color.

If more than one resource changes status at the same time, SDF shows the information for the system with the highest priority status first. See “Subsystem Colors and Priorities” on page 135 for information on status priorities. You can press PF8 to page through the Detail Status Display panels for other resources on KEY1.

Monitoring with the Status Display Facility (SDF)

For example, in the Best Goods Company, you move the cursor to MVSESA3 (the system in red). When you press PF2, you see the Detail Status Display panel for DB2, which is the resource with the highest priority, because it is BROKEN.

```
----- DETAIL STATUS DISPLAY -----
                                         1 OF 514

COMPONENT: HUGC2APL0          SYSTEM   : KEY1
COLOR   : RED                PRIORITY : 120
DATE    : 05/17/04          TIME     : 10:11:42
REPORTER : GATIPUFM         NODE      : IPUFM

REFERENCE VALUE: HUGC2APL0

AOF571I 10:11:42 : KEY1.HUGC2APL0 SUBSYSTEM STATUS FOR JOB HUGC2A0 IS
BROKEN

===>
1=HELP 3=RETURN 4=DELETE 6=ROLL 7=UP 8=DOWN 9=ASSIST 11=BOTTOM 12=TOP
```

Figure 56. Example SDF Detail Status Display Panel for HUGC2APL0

This panel also shows the automation or human operator that sent the notification and gives the priority number that corresponds to the automation status. Priority numbers determine which color the resource is displayed in and the order of the Detail Status Display panels (the lower the number the higher the priority).

In this example, the panel is one of four Detail Status Display panels (as indicated at the top right corner of the panel), one for each of the resources under MVSESA3 that has detail information. Press PF8 to see the Detail Status Displays for the other resources. PF8 pages through the displays in order of descending status priority.

To return to a particular panel, you can press PF7, which pages through the displays in order of ascending priority, or you can press PF12, which takes you back to the first Detail Status Display in the stack, the one with the highest priority status.

PF3 Returns to the system panel.

PF4 Deletes the panel you are currently on. (The record is physically deleted.)

Note: You should normally not use this PF key.

PF9 Takes you to the Operator Assist panel when in assist mode.

PF11 Moves to the Detail Status Display panel for the resource with the lowest priority, the last panel in the Detail Status Display stack.

PF12 Moves to the Detail Status Display panel for the resource with the highest priority, the first panel in the Detail Status Display stack.

Note: PF keys can be customized. How to achieve this is described in the *System Automation for z/OS Programmer's Reference*.

Monitoring with the Status Display Facility (SDF)

How to Look at the Resource Hierarchy

To look at the resource hierarchy, select a resource with the cursor and press PF8 to go down a level or PF7 to go up a level.

```
Resource Status for System Boulder1

System  Subsystem  WTORs    Gateways  Spool    MVS Comps
FOC1    SDSF        NetView
FOC2    AOCAPPL3
TGT1    AOCSS1
TGT3    JES
SYS1    VTAM
SYS2    TSO
SYS4    RMF
```

How to Monitor WTORs with SDF

To view a WTOR through SDF (if SDF is set up to display WTORs) you can type SDF on a NetView console to use the status display facility. Position the cursor on a job under WTORs (for example, INGESO, in the following panel extract) and press PF2 for more details. A Detail Status Display panel will be shown, containing text and information about an outstanding WTOR message.

```
Resource Status for System Boulder1

System      Subsystem  WTORs      Gateways  Spool    MVS Comps
SYS_TSCF2   INGESO     INGESO

```

How to Monitor Application Groups with SDF

To view application groups through SDF, - if SDF is set up to display application groups (see also "How SDF Works" on page 137 - , you can type SDF on a NetView console to use the status display facility. By using PF8 (ZOOM), navigate to the SUMMARY STATUS panel for the system, for which you want to monitor application groups. In our customized example from Figure 57, we selected system **KEY3**.

```
KEY3

                                SUMMARY-STATUS
                                KEY3

                                >WTOR      >GATEWAY
                                >MESSAGES >MVSCOMP

                                !-----!

                                >MVS

                                >SYSTEM

                                >APPLS    >GROUPS

                                >TEST

                                06/05/04 14:02

====>
1=HELP 2=DETAIL 3=RETURN 6=ROLL 7=UP    8=ZOOM 12=TOP
```

Figure 57. SDF Summary Status Panel

Monitoring with the Status Display Facility (SDF)

Now position the cursor under **GROUPS** and press PF8. In our example, we will see a customized panel containing information about application groups on **KEY3**.

```
KEY3                                KEY3:  APPLGROUP-STATUS

QFETCH

SYBNET_G

SYBSYS_G

                                06/05/04 14:13
===>                                13=EXPLAIN          17=START 18=STOP
1=HELP 2=DETAIL 3=RETURN 6=ROLL 12=TOP 20=APPLGR 21=ACF 22=ASF 23=LIST 24=INFO
```

Positioning the cursor on an application group of your choice and pressing PF2 (DETAIL) will display the SDF Detail Status Display panel for the selected application group, in our example, **QFETCH** (see Figure 58).

```
----- DETAIL STATUS DISPLAY -----                                1 OF 3

COMPONENT: QFETCH                                SYSTEM   : KEY3
COLOR    : PINK                                PRIORITY :    490
DATE     : 06/05/04                            TIME     : 12:57:08
REPORTER : AUTMSG                                NODE     : IPSNO

REFERENCE VALUE: QFETCH

QFETCH GROUP STATUS FOR SYSTEM APPLGROUP IS CTLDOWN

===>
1=HELP    3=RETURN    6=ROLL 7=UP 8=DOWN 9=ASSIST 10=QUIT 11=BOTTOM 12=TOP
```

Figure 58. SDF Detail Status Display Panel for an Application Group

How to Monitor and Control a Sysplex

SA z/OS provides a command interface to monitor and operate sysplexes. It helps operators to manage critical sysplex resources like coupling facilities and CF structures, couple data sets, and more. You can benefit from reduced sysplex operations complexity leading to higher sysplex availability and greater productivity.

The following section describes the panels of the system operations commands that support you in working with sysplex specific automation tasks. For details about the syntax and parameters of these commands, refer to *System Automation for z/OS Operator's Commands*.

How to Work with INGPLEX

SA z/OS provides the command INGPLEX to monitor and control the couple data sets and coupling facilities of your sysplex.

How to Monitor and Control a Sysplex

If you enter INGPLEX in the command line without any parameters, the following selection panel is displayed:

```
INGLX000          SA z/OS - Command Dialogs
Domain Id = IPSFM  ----- INGPLEX -----          Date = 05/20/04
Operator Id = HIR                                     Time = 23:26:27

Sysplex . . . . . : KEY1PLEX

Select the desired command:                          INGPLEX ...

  1 Display systems (including ETR & signalling paths)  SYStem
  2 Display consoles                                   CONsole
  3 Control coupling facilities                        CF
  4 Control couple data sets                          CDS

  6 Display IPL information                            IPL
  7 Control dumps                                     DUMP
  8 Health Checker best practices                     BESTpractices
  9 Health Checker results

Command ==>
F1=Help      F2=End      F3=Return
F6=Roll
F12=Retrieve
```

Figure 59. INGPLEX Selection Panel

Specify the number or the function and press Enter.

These subcommands support the following functions:

- SYStem** Displays the target sysplex name, its GRS mode and its systems. For each system, details about the system name and status are shown.
- CONsole** Displays details about the target sysplex. For example, the master console name, WTO and WTOR buffer utilization, and the number of queued messages.
- CF** Displays the coupling facilities in the sysplex. For each coupling facility the structures and paths can be displayed, and the coupling facility can be drained and enabled.
- CDS** Displays the couple data sets in the sysplex. For each couple data set you can obtain detailed information, and you can switch the couple data sets or allocate new alternate couple data sets.
- IPL** Collects IPL information.
- DUMP** Displays the DUMP submenu.
- BESTpractices** Displays information about the currently active HealthChecker best practices. This information is retrieved from the system that performs the global and local checks.

These functions are described in more detail in the following sections.

INGPLEX SYStem

INGPLEX SYStem displays the target sysplex name, its GRS mode and its member systems.

```

AOFKX100          SA z/OS - Command Dialogs          Line 1   of 4
Domain ID   = IPSFO      ----- INGPLEX SYSTEM -----   Date = 10/02/01
Operator ID = OPER1                                Time = 16:11:29

Sysplex . . . . . : KEY1PLEX
GRS Mode . . . . . : STAR

Display more info: C CPU E ETR I IPL O IOS S STOR/ESTOR
Signalling Path : D device T structure

Monitor
----- SSUM -----
Cmd  System  Status  Timestamp  INTERVAL  Action  TIME  WEIGHT
---  -
-    KEY1    ACTIVE  16:11:25  86400     PROMPT  86400  1
-    KEY2    ACTIVE  16:11:25  86400     PROMPT  86400  1
-    KEY3    ACTIVE  16:11:27  86400     PROMPT  86400  1
-    KEY4    ACTIVE  16:11:23  86400     ISOLATE  0       1

Command ==>
F1=Help      F2=End      F3=Return   F6=Roll
              F9=Refresh  F12=Retrieve
    
```

Figure 60. INGPLEX SYStem Panel

- The **Sysplex** field shows the name of the sysplex.
- The **GRS Mode** field shows the GRS mode of the target system. The mode can be either STAR or RING.
- The **CMD** column allows you to specify one of the command codes to display more information. Type the appropriate letter next to the resource name, and press Enter.
- The **System** column shows the name of the system.
- The **Status** column shows the status of the system.
- The **Monitor Timestamp** column shows the last time stamp recorded for status monitoring on this system.
- The **INTERVAL** column shows the system failure detection interval in seconds. This interval is the time XCF lets elapse without a status update before assuming that the system failed.

The last three columns contain configuration data of the SFM policy (if applicable).

- The SSUM Action field shows the SSUM action. It can be one of the following:
 - ISOLATE
 - DEACTIVATE
 - RESET
 - PROMPT
 - N/A
- The SSUM TIME field shows the SSUM interval as specified in the current SFM policy.
- The SSUM WEIGHT field shows the SSUM weight specified in the current SFM policy. This weight is used in sysplex configurations after a signalling connectivity failure.

To display more detailed information, enter the following command codes:

How to Monitor and Control a Sysplex

- C** Displays the online or offline status of one or more processors and any vector facilities, or ICRFs attached to those processors
- E** Displays the timer synchronization mode and ETR ports
- I** Displays IPL information
- O** Displays IOS-related configuration information
- S** Displays the number of megabytes of central and expanded storage assigned and available to the system
- D** Displays the device number of one or more inbound or outbound signalling paths that XCF can use and information about inbound or outbound XCF signalling paths to this system
- T** Displays detailed signalling path information for all coupling facility structures

INGPLEX CONsole

INGPLEX CONsole displays the following information for the target sysplex:

- The name of the master console
- WTO & WTOR buffer utilization
- Number of queued messages (replies) of various types
- Awaiting mounts
- Operator requests and list of consoles (name, status, authority, number of WTOR buffers, UD, device, system, ALTGRP, MSCOPE)

```

INGLX400          SA z/OS - Command Dialogs          Line 1 of 6
Domain Id = IPSF0  ----- INGPLEX CONSOLE -----          Date = 10/02/01
Operator ID = OPER1                               Time = 16:13:35

Sysplex . . . . . : KEY1PLEX          Master Console . . . : --none--
Message Buffer Usage : 10 / 9999      Reply Buffer Usage . : 10 / 99
Awaiting Replies . . : 10             Eventual Action . . : 0
Immediate Action . . : 0              Awaiting Mounts . . : 0
Critical Action . . : 3               Operator Requests . . : 0
-----

Cmds: D Details / R Requests

  Console  Status  AUTH  NBUF  UD  Device  System  ALTGRP  MSCOPE
  -----  -----  -----  -----  --  -----  -----  -----  -----
- MASTER  INACTIVE  MASTER  n/a  Y  -none-  --none--  --none--  *ALL
- MASTER1 INACTIVE  ALL    n/a  N  -none-  --none--  --none--  *ALL
- 03     INACTIVE  MASTER  n/a  N  -none-  --none--  --none--  *ALL
- 04     INACTIVE  MASTER  n/a  N  -none-  --none--  --none--  *ALL
- 05     INACTIVE  MASTER  n/a  N  -none-  --none--  --none--  *ALL
- 06     INACTIVE  MASTER  n/a  N  -none-  --none--  --none--  *ALL

Command ==>
F1=Help      F2=End      F3=Return   F6=Roll
              F9=Refresh   F12=Retrieve
  
```

Figure 61. INGPLEX Console Panel

- The **Sysplex** field shows the name of the sysplex.
- The **Message Buffer Usage** field shows the limit of the number of WTO message buffers allowed outstanding.
- The **Awaiting Replies** field shows a decimal number of messages awaiting replies.

- The **Immediate Action** field shows a decimal number of outstanding immediate action messages (with descriptor codes 1 or 2). If the number is greater than 99999, asterisks appear in this field.
- The **Critical Action** field shows a decimal number of outstanding critical eventual action messages (with descriptor code 11). If the number is greater than 99999, asterisks appear in this field.
- The **Master Console** field shows the name of the master console.
- The **Reply Buffer Usage** field shows the limit of the number of WTOR message buffers allowed outstanding. The maximum value of *yyyy* is specified by the RMAX parameter in the CONSOLxx parmlib member.
- The **Eventual Action** field shows a decimal number of outstanding eventual action messages (with descriptor code 3). If the number is greater than 99999, asterisks appear in this field.
- The **Awaiting Mounts** field shows a decimal number of outstanding mount requests.
- The **Operator Requests** field shows a decimal number of outstanding requests for operator intervention.
- The **CMD** column lets you specify the command codes shown on the panel. Type the appropriate letter next to the resource name, and press Enter.
- The **Console** column shows the name of the console as specified in the CONSOLxx parmlib member.
- The **Status** field shows the status of the console. The following values can occur:

HARDCOPY	Hardcopy log. This condition is indicated when the console is active on the system where the command processes.
ACTIVE	Active console.
ACTIVE-P	In the process of becoming an active console. This condition will be indicated only when the console is active on the system where the command is processing.
MASTER	Master console
INACTIVE	Inactive console
INACT-P	In the process of becoming a non-active console. This condition will be indicated only when the console is active on the system where the command is processing.
PROB-DET	The active system console is in the problem determination mode. PD is indicated only for the system console.
SUBSYS	Subsystem-allocatable console
- The **AUTH** column shows which commands can be entered from this console. The following values can occur:

ALL	Any INFO SYS, IQ or CONS command can be entered from this console.
CONS	INFO commands and any commands from the console command group can be entered from this console.
INFO	Any command from the informational command group can be entered from this console.
IO	INFO commands and any commands from the I/O Control command group can be entered from this console.

How to Monitor and Control a Sysplex

MASTER The specified console is authorized to enter any operator command.

NONE This console has no command authority.

SYS INFO commands and any commands from the system control command group can be entered from this console.

- The **NBUF** column shows the number of WTO message buffers currently queued to this console. If *nnnn* is greater than 9999, asterisks (****) appear in this field.
- The **UD** column shows whether this console is receiving messages with the UD attribute.
- The **Device** column shows the name of the console as specified in the CONSOLxx parmlib member.
- The **System** column shows the system name of the active console.
- The **ALTGRP** column shows the alternate group defined for this console.
- The **MSCOPE** column lists the name of the system or systems from which this console is receiving unsolicited messages. Note that these systems might be different from the system where this console is physically attached.

INGPLEX CF

With INGPlex CF you can:

- Drain or enable a selected coupling facility
- Display the paths and structures to a selected coupling facility

```

INGLX900          SA z/OS - Command Dialogs          Line 1 of 2
Domain Id = IPSFO ----- INGCF -----          Date = 10/02/01
Operator ID = OPER1                               Time = 16:14:27

Sysplex . . . . . : KEY1PLEX          SM process level . : 9
-----
Cmds: D drain CF / E enable CF / P display sender paths / S display structures

  CF Name   Total Space   Free Space   Free%   Volatile   CF level
  -----   -
- CF01      509184 K      289792 K   56.91   YES        9
- CF02      509184 K      508672 K   99.90   YES        9

Command ==>
F1=Help      F2=End      F3=Return   F6=Roll
              F9=Refresh   F12=Retrieve
  
```

Figure 62. INGCF Panel

This panel displays all coupling facilities in a sysplex including particular storage information of each coupling facility. On this panel you can select one of the following actions by entering the command code next to the coupling facility:

D (Drain CF)

Releases a coupling facility from its structures and connections in order to remove it from the sysplex for maintenance.

E (Enable CF)

Integrates a coupling facility into a sysplex.

P (Display Sender Paths)

Displays the paths from the systems in the sysplex having a connection defined to the specified coupling facility. For each path the physical and the logical status are displayed.

S (Display Structures)

Shows all structures allocated in the specified coupling facility.

These actions are described in detail in the following sections.

Drain CF: With this function, you can remove a CF from the sysplex, for example, for maintenance purposes. You can perform the following sequence of tasks:

1. Display information for all allocated structures of the CF.
2. Rebuild all rebuildable structures on another CF and delete instances of structures on the target CF that are being duplexed on another CF.
3. Force the deletion of structures that have no active connectors and cannot be rebuilt.

Note that there are structures that you can neither rebuild nor delete with the force action. These include the structures that have at least one active connector and do not support rebuild. To remove such structures first disconnect all active connectors, and then delete the structure manually if it is persistent or has persistent connections.

4. Disconnect the CF from the systems with which it is connected

The DRAIN option ensures that these actions are performed in the correct order, as specified above.

Depending on the status of the CF and its allocated structures, you can execute one of four actions with the program function keys:

REBUILD (PF10)

Starts the rebuild of structures that can be rebuilt on another CF. Therefore, a rebuild is only initiated for structures whose preference list contains more than one CF.

There are two methods for rebuild, user-managed and system-managed rebuild. User-managed rebuild is supported for all release levels.

System-managed rebuild is only available with systems that have z/OS 2.8 or above; it must have been enabled by formatting the CFRM couple data sets with the specification

```
ITEM NAME(SMREBLD) NUMBER(1)
```

System-managed rebuild is only performed when the requirements for user-managed rebuild are not met. This applies, for example, to structures without active connectors.

The REBUILD action also deletes all structure instances on the target CF that are being duplexed on another CF.

FORCE (F5)

Forces the deallocation of structures with one of the following conditions:

- No connection exists.
- No alternate CF for structure with no active connections.
- No alternate CF for structure with no connections.

This action is only available after all structures that can be rebuilt have been rebuilt.

How to Monitor and Control a Sysplex

DRAIN (F4)

Disconnects the coupling facility from its connected systems by setting the sender path(s) OFFLINE.

This action is only enabled after all structures of the target CF have been removed to another CF or deallocated. Note that structures that have active connectors but do not support rebuild cannot be removed with PF10 or F5. They must be deallocated manually before execution of this step is enabled.

Example of draining a CF:

```
INGLX901          SA z/OS - Command Dialogs          Line 1 of 31
Domain Id = IPSFO ----- INGCF DRAIN -----          Date = 10/02/01
Operator Id = OPER -----                          Time = 07:01:00

Coupling Facility ==> CF01          Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . : ALL
-----
Structure          Condition
-----
DFHXQLS_TESTTSQ1  No active connections. System-managed rebuild supported.
DSNG_LOCK1
DSNG_SCA          System-managed rebuild is supported.
ISGLÖCK
ISTGENERIC
IXCGRS
IXCplex_PATH1
M7SG_LOCK1        *No alternate CF for structure with no active connections.
M7SGEMHQ          No active connections. System-managed rebuild supported.
M7SGMSGQ          No active connections. System-managed rebuild supported.
M7SGMSGQOV        No active connections. System-managed rebuild supported.

Command ==>
F1=Help          F2=End          F3=Return          F6=Roll
F8=Forward       F9=Refresh      PF10=Rebuild      F12=Retrieve
```

Figure 63. DRAIN CF Command Dialog

The status of the CF (NORMAL) and the authorization type of the operator (ALL) are displayed on the right side of the panel. The main part of the panel consists of a list containing the structures allocated in CF1 and their conditions. The conditions are classified by color and an asterisk. The asterisk signifies that a structure cannot be rebuilt.

The only action that is enabled is REBUILD with PF10. Pressing PF10 calls the following confirmation panel:


```

INGLX92R          SA z/OS - Command Dialogs
Domain Id   = IPSFO   ----- INGCF DRAIN -----
Operator ID = OPER1
Date = 10/02/01
Time = 07:01:04

Coupling Facility . : CF01
Sysplex . . . . . : KEY1PLEX

                R E B U I L D Confirmation

The REBUILD process runs asynchronously on the next system in the sysplex that
has access to the CFRM couple data set and can perform all necessary actions.
Each structure that has no * indicator in front of its status is rebuilt to its
status accordingly. The structures are processed in sequence. Once started use
the refresh PF key for getting the current status of the process. When more
than one structure is being rebuilt a timeout occurred indicating that XCF is
very busy. But processing continues. A display without any structure or only
structures that cannot be rebuilt indicates a successful completion.

Command ==>
                F2=End          F3=Return          F6=Ro11
                PF10=Go        F11=Cancel    F12=Retrieve
    
```

Figure 64. DRAIN Command Dialog: Confirmation Panel for REBUILD

After PF10 has been pressed, the structures are in the process of being rebuilt:

```

INGLX901          SA z/OS - Command Dialogs          Line 1 of 6
Domain Id   = IPSFO   ----- INGCF DRAIN -----
Operator ID = OPER1
Date = 10/02/01
Time = 07:18:44

Coupling Facility ==> CF01          Status . . . . . : REBUILDING
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . : ALL
-----
Structure      Condition
-----
DSNG_LOCK1     Structure is awaiting rebuild.
ISGLOCK        Structure is awaiting rebuild.
ISTGENERIC     Structure is awaiting rebuild.
IXCGRS         Structure is being rebuilt.

Command ==>
F1=Help        F2=End          F3=Return          F6=Ro11
                F9=Refresh          F12=Retrieve
    
```

Figure 65. DRAIN Command Dialog: Rebuild Process

After the rebuild has been completed, a panel similar to the following is displayed:

How to Monitor and Control a Sysplex

```

INGLX901          SA z/OS - Command Dialogs      Line 1 of 1
Domain Id   = IPSFO      ----- INGCF DRAIN -----      Date = 10/02/01
Operator ID = OPER1                                           Time = 07:53:36

Coupling Facility ==> CF01          Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure      Condition
-----
M7SG_LOCK1     *No alternate CF for structure with no active connections.

Command ==>
F1=Help      F2=End          F3=Return      F5=Force      F6=Roll
              F9=Refresh                                F12=Retrieve

```

Figure 66. DRAIN Command Dialog: Panel after Rebuild

One structure could not be rebuilt because no alternate CF is specified in its preference list. The REBUILD status is no longer available. Instead, the FORCE action (F5) is offered because the structure that could not be rebuilt has a condition that allows forcing the deallocation of the structure. Pressing F5 displays a confirmation panel similar to that for REBUILD. Pressing PF10 on the confirmation panel and refreshing the command dialog after the action has been completed results in the following panel:

```

INGLX901          SA z/OS - Command Dialogs      Line
Domain Id   = IPSFO      ----- INGCF DRAIN -----      Date = 10/02/01
Operator ID = OPER1                                           Time = 08:12:28

Coupling Facility ==> CF01          Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure      Condition
-----

Command ==>
F1=Help      F2=End          F3=Return      F4=Drain      F6=Roll
              F9=Refresh                                F12=Retrieve

```

Figure 67. DRAIN Command Dialog: Panel after Forcing

No more structures are allocated in the coupling facility, so that the coupling facility can be released from the connections with the systems of the sysplex. Consequently, INGCF DRAIN enables the DRAIN action (F4). After completion of

that action, the status of the coupling facility changes to DRAINED, as shown on the following panel:

```

INGLX901          SA z/OS - Command Dialogs          Line
Domain Id  = IPSFO  ----- INGCF DRAIN -----      Date = 10/02/01
Operator ID = OPER1                                     Time = 08:12:32

Coupling Facility ==> CF01          Status . . . . . : DRAINED
Sysplex . . . . . ==> KEYIPLEX      Permission . . . . . : ALL
-----
Structure          Condition
-----

Command ==>
F1=Help           F2=End           F3=Return         F6=Ro11
                  F9=Refresh         F11=Shutdown    F12=Retrieve
    
```

Figure 68. DRAIN Command Dialog: Panel after Draining

Because the coupling facility is no longer connected to any system, it can be inactivated. After pressing F11 the status of the coupling facility changes to INACTIVE:

```

INGLX901          SA z/OS - Command Dialogs          Line
Domain Id  = IPSFO  ----- INGCF DRAIN -----      Date = 02/13/01
Operator Id = NETOP1                                     Time = 08:12:32

Coupling Facility ==> CF01          Status . . . . . : INACTIVE
Sysplex . . . . . ==> KEYIPLEX      Permission . . . . . : ALL
-----
Structure          Condition
-----

Command ==>
F1=Help           F2=End           F3=Return         F6=Ro11
                  F9=Refresh         F12=Retrieve
    
```

Figure 69. DRAIN Command Dialog: Panel after Inactivation

Enable CF: The ENABLE function of the INGCF command is intended to support the integration *and* reintegration of a coupling facility into a sysplex. With this option, you can:

1. Activate the target coupling facility.

How to Monitor and Control a Sysplex

2. Connect the systems of the sysplex with the coupling facility.
3. Switch to another CFRM policy if the target CF is not defined in the active policy and a suitable policy is available.
A suitable CFRM policy must contain:
 - A definition of the target CF
 - Appropriate definitions for every active CF and every allocated structure
4. Rebuild all structures on the target CF whose preference list starts with this CF, unless this is excluded by other requirements.

The ENABLE CF option ensures that these actions are performed in the correct order, as specified above.

The following function keys support these actions:

***ACTIVATE (F11)**

This action activates the CFCC (Coupling Facility Control Code) through the BCP Internal Interface by an ACTIVATE command.

Note: This function key is unavailable when running on a z/OS image which runs under z/VM®.

***ENABLE (F4)**

Sets the sender paths of all systems of the sysplex to ONLINE. This action is enabled when the CF is active.

***SWITCH (F5)**

Switches to another CFRM policy when the target CF is not defined in the active CFRM policy and a suitable policy is available. When there is more than one suitable policy you can choose one of these from a selection panel.

A CFRM policy is suitable when it contains:

- A definition of the target CF
- Definitions for every active CF and every allocated structure

This action is only available when the target CF is active, but not defined in the current CFRM policy.

***POPULATE (PF10)**

Starts a rebuild process in which all structures that have the target CF at the beginning of their preference list but are currently allocated on another CF are allocated on the target CF.

This action requires that the CF is enabled, connected to all members of the sysplex, and defined in the current CFRM policy. The action is offered whenever INGCF ENABLE detects that a structure is not allocated on the target CF although it is the preferred CF of that structure.

Note: When you have drained a coupling facility with INGCF DRAIN and then reintegrate it with INGCF ENABLE, be aware that the set of structures that are allocated on the target coupling facility after population can be different from the original set before the draining. Typically, this happens when the original set does not contain exactly those structures that have the target coupling facility at the first position in their preference list.

Note that these actions can only be performed when INGCF ENABLE is called in full mode. In line mode, only the display function is available.

In the following example, a coupling facility that has already been activated is reintegrated into the sysplex in two steps:

1. The coupling facility is connected to all systems of the sysplex.
2. All structures that have the target coupling facility as the first coupling facility in their preference list are allocated on the target coupling facility.

If you issue INGCF with the option ENABLE, you can specify the coupling facility to be reintegrated, for example by entering INGCF ENABLE CF02. In this case, the panel of Figure 70 is displayed at once. If you do not specify a coupling facility name, INGCF shows a selection panel with all coupling facilities that are defined in the sysplex. After selection of CF02, INGCF displays the following panel:

```

INGLX901          SA z/OS - Command Dialogs          Line
Domain Id   = IPSFO      ----- INGCF ENABLE -----      Date = 10/02/01
Operator ID = OPER1                                           Time = 11:06:06

Coupling Facility ==> CF02          Status . . . . . : DRAINED
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure      Condition
-----
-----

Command ==>
F1=Help      F2=End          F3=Return    F4=Enable          F6=Ro11
              F9=Refresh          F12=Retrieve
  
```

Figure 70. ENABLE Command Dialog: Panel before any Action

The selected CF has already been activated manually, therefore its status, as shown on the right side of the panel, is DRAINED. The authorization type of the operator (ALL) is also displayed on the right side of the panel. The main part of the panel is empty because no structures are allocated in CF02. The only action that is activated is ENABLE with PF4. If you press PF4 the following confirmation panel is displayed:

How to Monitor and Control a Sysplex

```

INGLX92E          SA z/OS - Command Dialogs          Date = 10/01/01
Domain Id = IPSFO  ----- INGCF ENABLE -----      Time = 11:06:20
Operator ID = OPER1

Coupling Facility . : CF02
Sysplex . . . . . : KEY1PLEX

                E N A B L E Confirmation

The ENABLE process runs asynchronously on the next system in the Sysplex that
has access to the CFRM couple data set. All sender paths of all system in the
sysplex are set to ONLINE. Once started use the refresh PF key for getting
the current status of the process. The status NORMAL indicates a successful
completion.

Command ==>
                F2=End      F3=Return      F6=Roll
                PF10=Go     F11=Cancel   F12=Retrieve
    
```

Figure 71. Confirmation Panel for ENABLE

After pressing PF10 on the confirmation panel, the command dialog changes as follows:

```

INGLX901          SA z/OS - Command Dialogs          Line 1 of 3
Domain Id = IPSFO  ----- INGCF ENABLE -----      Date = 10/02/01
Operator ID = OPER1                                     Time = 11:06:39

Coupling Facility ==> CF02          Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . : ALL
-----
Structure          Condition
-----
GRPYCSQ_ADMIN     Structure is currently allocated in CF01.
GRPYHSAQUEUE      Structure is currently allocated in CF01.
HSA_LOG           Structure is currently allocated in CF01.

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
F9=Refresh   PF10=Populate  F12=Retrieve
    
```

Figure 72. ENABLE Command Dialog: Panel after Enabling

The status has changed to NORMAL, and you can populate the CF with PF10. This implies that the target CF is defined in the active CFRM policy.

The structure list contains three entries with the condition 'Structure is currently allocated in CF01.'. These are the structures that are currently allocated in CF01, but have CF02 at the first position in their preference list.

Pressing PF10 now populates the CF, and the refreshed panel looks similar to the following:

```

INGLX901          SA z/OS - Command Dialogs      Line 1 of 3
Domain Id = IPSFO  ----- INGCF ENABLE -----   Date = 10/02/01
Operator ID = OPER1                               Time = 11:17:35

Coupling Facility ==> CF02          Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure          Condition
-----
GRPYCSQ_ADMIN      System-managed rebuild is supported.
GRPYHSAQUEUE       System-managed rebuild is supported.
HSA_LOG

Command ==>
F1=Help           F2=End           F3=Return          F6=Ro11
                  F9=Refresh          F12=Retrieve

```

Figure 73. ENABLE Command Dialog: Panel after Populating

The POPULATE action is no longer available because all structures whose preference list starts with CF02 are now allocated in CF02.

Display Sender Paths: With this function, the paths from the connected systems to the specified coupling facility are displayed.

The last sender path of each system can only be set to OFFLINE when no more structures are allocated.

```

INGLX903          SA z/OS - Command Dialogs      Line 1 of 4
Domain Id = IPSFN  ----- INGCF PATH -----   Date = 06/20/01
Operator ID = HIR                               Time = 10:28:49

Coupling Facility ==> CF01          Allocated Structures: 37
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Cmds: F set OFFLINE / N set ONLINE

  System  CHPID  Physical          Logical  Type
-----
- KEY1    A5     ONLINE           ONLINE  CFS
-         A9     ONLINE           ONLINE  CFS
- KEY2    A5     ONLINE           ONLINE  CFS
-         A9     ONLINE           ONLINE  CFS
- KEY3    A5     ONLINE           ONLINE  CFS
-         A9     ONLINE           ONLINE  CFS
- KEY4    05     ONLINE           ONLINE  CFS
- KEY6    A5     ONLINE           ONLINE  CFS
-         A9     ONLINE           ONLINE  CFS
-

Command ==>
F1=Help           F2=End           F3=Return          F6=Ro11
                  F8=Forward       F9=Refresh          F12=Retrieve

```

Figure 74. Display Sender Path Panel

The following command codes are available:

- F** Sets the sender path OFFLINE.
- N** Sets the sender path ONLINE.

How to Monitor and Control a Sysplex

- If you have issued INGCF with the PATH parameter, the **Coupling Facility** field is an input field. To display the path list of another coupling facility specify the name of the coupling facility in this field and press Enter.
- The **Allocated Structures** field shows the number of allocated structures.
- The **Permission** field shows your authorization level.
- The **System** column contains the names of the systems that are connected to the target CF.
- The **CHPID** column shows the IDs of the sender channel paths.
- The **Physical** column shows the status of the sender channel paths.
- The **Logical** column shows the logical status of the paths to that CF.
- The **Type** column shows the type of the sender channel paths.

Display Structures: The STRUCTURE function of the INGCF displays the allocated structures of a coupling facility. You can initiate a rebuild or deallocation of a selected structure if the conditions for these actions are satisfied.

```

INGLX904          SA z/OS - Command Dialogs          Line 1 of 3
Domain ID = IPSFM  ----- INGCF STRUCTURE -----   Date = 02/22/04
Operator ID = NETOP1                               Time = 16:09:04

Coupling Facility ==> CF01
Sysplex . . . . . ==> KEYIPLX          Permission . . . . : ALL
Include condition ==> YES (Yes/No - Condition retrieval takes longer)
-----
Cmds: D display details / F force / P stop duplex / R rebuild / S start duplex

  Structure      P D Condition
-----
- DSNB_GBP0      U Duplex rebuild is active.
- DSNB_GBP32K    P U
- DSNB_LOCK1     S Duplex rebuild is active.
- DSNB_SCA       S System-managed rebuild is supported.
- ISGLOCK
- ISTGENERIC     System-managed rebuild is supported.
- IXCGRS
- IXCVLF
- M8SGEMHQ       System-managed rebuild is supported.
- M8SGMSGSL
- M8SGMSGQ       System-managed rebuild is supported.
- RRSARCH
- RRSDELAY
- RRSMAIN
- RRSREST        P *No alternate CF defined or available.

Command ==>
F1=Help      F2=End      F3=Return    F6=Roll
              F9=Refresh   F12=Retrieve

```

Figure 75. Display Structures Panel

- If you have specified INGCF with the STR parameter, the **Coupling Facility** field is an input field. To display the structure list of another coupling facility, specify the name of the coupling facility in this field and press Enter.
- The **Include Condition** field is an input field. By entering Yes or No in this field you determine whether or not the conditions of the structures are displayed in the **Structure** column.
- The **Permission** field shows your authorization level. There are two possible values, ALL and DISPLAY. DISPLAY indicates that you can only use the display functions. ALL indicates that you can also rebuild and delete structures.

You can specify an action code before every structure entry. The codes you can enter depend on your authorization level

- The **Structure** column shows the names of the structures.
- The letter P in the P column indicates that policy changes are pending for the structure.

A structure has policy changes pending when it was allocated at the time of a CFRM policy switch, and XES could not bring the switch into effect for that structure. One reason for a pending policy change is that the old and the new policy define the structure differently, for example, with different preference lists.

- The **Condition** column shows the status of the structures. You can switch the display of the conditions on and off with the **Include Condition** field.
- The D field indicates the type of duplexing that is possible. The following values are possible:

U	User-managed duplexing
S	System-managed duplexing
B	User-managed and system-managed duplexing

The following command codes are available:

- D** Displays detail information about the structure.
- *F** Forces the deallocation of the structure if it has one of the following conditions:
 - No connection exists.
 - No alternate CF for structure with no active connections.
 - No alternate CF for structure with no connections.

When you try to force the deallocation of a structure that can be rebuilt, an error message is issued.

- *P** Stops duplexing of the selected structure.
- *R** Starts the rebuild of the selected structure. Depending on the PENDING status, the automation starts the rebuild with a different LOCATION parameter (PENDING uses the parameter LOCATION=NORMAL, otherwise LOCATION=OTHER). A rebuild with the parameter LOCATION=OTHER is only initiated for structures whose preference list contains more than one coupling facility.

There are two methods for rebuild, user-managed and system-managed rebuild. User-managed rebuild is supported for all release levels. System-managed rebuild is only available with systems that have z/OS 2.8 or above. It must have been enabled by formatting the CFRM couple data sets with the specification

```
ITEM NAME(SMREBLD) NUMBER(1)
```

System-managed rebuild is only performed when the requirements for user-managed rebuild are not met. This applies, for example, to structures without active connectors.

INGCF STRUCTURE accepts a rebuild request for structures with the condition 'No active connection exists.', but *deallocates* them. See the section "Structure Conditions" in the description of the INGC command in *System Automation for z/OS Operator's Commands*.

How to Monitor and Control a Sysplex

The rebuild function works differently depending on the status of the structure (PENDING calls LOCATION=NORMAL, otherwise LOCATION=OTHER).

*S Starts duplexing of the selected structure.

There are two methods for duplexing, user-managed and system-managed duplexing. User-managed duplexing is supported for all release levels. System-managed duplexing is only available when all systems in the Parallel Sysplex have been upgraded to z/OS 1.2 or later with APAR OW41617, and appropriate APARs listed in the CFDUPLEX PSP bucket (for more information, see *System-Managed CF Structure Duplexing*, GM13-0103-03). System-managed duplexing must have been enabled by formatting the CFRM couple data sets with the specification

```
ITEM NAME(SMDUPLEX) NUMBER(1)
```

System-managed duplexing is only performed when the requirements for user-managed duplexing are not met. This applies, for example, to structures without active connectors.

Starting the duplex rebuild of a structure requires at least the policy entry allowing the duplex rebuild of the structure. If there is no entry the duplex rebuild is disabled. The other requirements depend on the type of the duplex rebuild. When all connectors to a structure allow user-managed duplex rebuild, this type takes precedence over system-managed duplex rebuild. However, user-managed rebuild also requires at least one active connector. Thus, when the operator starts the duplex rebuild for a structure allowing user-managed duplex rebuild as well as system-managed rebuild but without having active connectors, XCF tries to initiate a system-managed duplex rebuild. System-managed duplex rebuild has the following requirements:

- System-managed rebuild must be supported by all connectors.
- The structure must be allocated in a coupling facility supporting system-managed duplexing and another coupling facility supporting system-managed duplexing must be defined in its preference list.
- The CFRM couple data set must support system-managed duplex rebuild and the structure must not have a policy change pending.
- The structure must be defined in the active CFRM policy when any connection state is not active.

INGPLEX CDS

The CDS function displays information about all the couple data sets in the system, including details of the corresponding policies. For every CDS type that is required by the implementation INGPLEX CDS allows the operator to:

- Switch from the primary to the alternate CDS
- Define a new alternate CDS
- Change the active policy (if applicable)

```

INGKX300          SA z/OS - Command Dialogs          Line 1 of 18
Domain ID = IPSFO ----- INGPLEX CDS -----      Date = 10/02/01
Operator ID = NETOP1          Sysplex = KEY1PLEX      Time = 16:15:20

System...: KEY3              Interval...: 86400          OPNotify: 86400
Maxmsg...: 999999           Cleanup...: 60             Retry...: 255
Classlen: 956               Max CFlevel: 9            COUPLExx: COUPLER1
SMREBLD.: N/A              Max SMlevel: 9

Cmds: A allocate alternate CDS / C display CHPIDs
      D display CDS information / P switch alternate CDS to primary CDS

Type      MS      Volume  Dev   Couple Dataset Name
-----
SYSPLEX
- PRIMARY...: 10  KEY1SP  260A  SYS1.KEY1.PXCFDCS
  ALTERNATE: 10  KEYLIB  2610  SYS1.KEY1.AXCFDCS
ARM
- PRIMARY...: 8   KEY1SP  260A  SYS1.KEY1.PARMCDS
  ALTERNATE: 8   KEYLIB  2610  SYS1.KEY1.AARMCDS
CFRM
- PRIMARY...: 32  KEY1SP  260A  SYS1.KEY1.PXESCDS
  ALTERNATE: 32  KEYLIB  2610  SYS1.KEY1.AXESCDS
LOGR
- PRIMARY...: 32  KEY1SP  260A  SYS1.KEY1.PLOGCDS
  ALTERNATE: 32  KEYLIB  2610  SYS1.KEY1.ALOGCDS

Command ==>
F1=Help      F2=End      F3=Return   F6=Ro11
              F8=Forward  F9=Refresh  F12=Retrieve
  
```

Figure 76. INGPLEX CDS Panel

This panel header contains sysplex-related information about the system on which the INGPLEX command was executed. The details are as follows:

- The **System** field shows the name of the system.
- The **Interval** field shows the system failure detection interval in seconds. This interval is the amount of time XCF lets elapse without a status update before assuming that the system failed.
- The **OPNotify** field shows the number of seconds that XCF waits before notifying the operator of a potential system problem.
- The **Maxmsg** field shows the default value for the maximum amount of kilobytes of message buffer space. This default value is used when MAXMSG is not specified on SETXCF START commands.
- The **Cleanup** field shows the number of seconds that XCF waits for cleanup of members.
- The **Retry** field shows the default value for the retry limit. This value is used when the RETRY keyword is not specified on SETXCF START commands.
- The **Classlen** field shows the default length (in bytes) of messages allowed for a transport class. This value is used when CLASSLEN is not specified on the SETXCF START CLASSDEF command.
- The **Max CFlevel** field shows the maximum CFLEVEL supported by this system. This system can connect to a coupling facility with a higher CFLEVEL than the value of **Max CFlevel** but would not be enabled to use any functions supported by the higher level coupling facility.
- The **COUPLExx** field shows the COUPLExx parmlib member used for system IPL.
- The **SMREBLD** field shows whether (value 1) or not (value 0) system-managed rebuild has been activated in the CFRM couple data set.

How to Monitor and Control a Sysplex

- The **Max SMlevel** field shows the maximum system-managed process level supported by this system.

For each couple data set you can enter one of the following command codes to initiate an action:

***A - allocate alternate CDS**

Replaces the current alternate CDS for a selected CDS type with a new one. There are two options how to do this:

- The alternate CDS is allocated automatically by SA z/OS.
This automatic allocation requires that spare volumes have been defined, and that one of these spare volumes is available. For details see *System Automation for z/OS Defining Automation Policy*.
- Specify the data set that is to be used as the new alternate CDS.
If you specify your own data set, observe the following:
 - The data set must exist
 - It must have been formatted with the XCF formatting tool
 - It must be at least as large as the current primary CDS, which means that every value you have passed to the XCF formatting tool (for example, in the case of a sysplex CDS, the maximum number of systems supported) must be equal to or greater than the corresponding value of the primary CDS.

C - display CHPIDs

Displays information about the channel paths for the selected CDS type.

D - display CDS information

Displays detailed information about the selected CDS type. This comprises the formatting parameters and the policies that are contained in the CDS, if applicable. When the CDSs of the selected type contain policies, the detail information panel provides further actions, namely:

display policy (D)

Displays details about the selected policy.

start policy (S)

Makes the selected policy the active policy. The policy switch must be confirmed before it is executed.

***P - switch alternate CDS to primary CDS**

This action makes the alternate CDS the primary. Since an alternate CDS is no longer available after the switch, SA z/OS displays a confirmation panel before the action is performed. On this panel you can specify a new alternate CDS. When CDS recovery is switched on and you do not supply your own alternate CDS, SA z/OS tries to allocate a new alternate CDS automatically. The special requirements for manual and automatic creation of the new alternate CDS are the same as those for the replacement of the alternate CDS (action code A).

Making an Alternate CDS the Primary CDS: In this example, the alternate LOGR couple data set is made the new primary CDS. A new alternate CDS is automatically generated.

To switch the LOGR couple data set, enter P before LOGR on the panel displayed in Figure 76 on page 159, and press Enter. INGPLEX CDS displays the following confirmation panel:

```

INGKX30A          SA z/OS - Command Dialogs
Domain ID  = IPSFO  ----- INGPLEX CDS -----      Date = 03/01/01
Operator Id = NETOP1      Sysplex = KEY1PLEX          Time = 10:08:13

                          SETXCF PSWITCH Confirmation

You are going to remove the LOGR   primary couple data set.
The alternate couple data set  SYS1.KEY1.ALOGCDS
becomes the primary as soon as you proceed with the GO function key.
Immediately after the switch, automation will try to allocate a new alternate
couple data set on one of the spare volumes defined during the customization.
If you want the automation to allocate your own alternate couple data set
complete the necessary information below.

Your alternate couple dataset...

Name   ==> _____
Volume ==> _____

Command ===>
          F2=End      F3=Return      PF10=Go      F6=Roll
          F11=Cancel  F12=Retrieve

```

Figure 77. Confirmation Panel for Switching from the Current Primary CDS to the Alternate CDS

Use this panel to determine how a new alternate CDS is to be created after the switch. You can either specify your own new alternate CDS or let SA z/OS create it for you. When you specify the new alternate CDS yourself, the data set must exist and must have been formatted with the XCF formatting tool. Automatic creation requires that spare volumes have been defined for LOGR couple data sets.

Pressing PF10 causes SA z/OS to generate the new alternate CDS. After returning to the CDS command dialog, refreshing the panel, and scrolling down with PF8, the panel looks as follows:

How to Monitor and Control a Sysplex

```

INGKX300          SA z/OS - Command Dialogs          Line 7   of 18
Domain ID = IPSFO  ----- INGPLEX CDS -----      Date = 03/01/01
Operator ID = NETOP1          Sysplex = KEY1PLEX      Time = 10:08:25

System..: KEY3              Interval...: 86400              OPNotify: 86400
Maxmsg..: 999999           Cleanup....: 60              Retry...: 255
Classlen: 956              Max CFlevel: 9              COUPLExx: COUPLER1
SMREBLD.: 1                Max SMlevel: 9

Cmds: A allocate alternate CDS / C display CHPIDs
      D display CDS information / P switch alternate CDS to primary CDS

  Type      MS   Volume Dev   Couple Dataset Name
  -----  --   -----  ---  -----
  -  CFRM
    PRIMARY...: 16  KEY1SP 260B  SYS1.KEY1.PXESCDS
    ALTERNATE: 16  KEYUSR 261C  SYS1.KEY1.AXESCDS
  -  LOGR
    PRIMARY...: 8   KEYUSR 261C  SYS1.KEY1.ALOGCDS
    ALTERNATE: 8   AOCUSR 262B  AOC.CDS.TEST.LOGR.CDS02

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
PF7=Back     PF8=Forward  F9=Refresh     F12=Retrieve
  
```

Figure 78. INGPLEX CDS Dialog Panel after the Switch

The previous alternate LOGR CDS has become the primary, and there is a new alternate, which was created by SA z/OS.

Switching the CFRM Policy: In this example, the active CFRM policy is switched.

Enter D before CFRM on the panel displayed in Figure 76 on page 159, and press Enter. The following panel is displayed:

```

INGKX311          SA z/OS - Command Dialogs          Line 1   of 5
Domain ID = IPSFO  ----- INGPLEX CDS -----      Date = 03/01/01
Operator ID = NETOP1          Sysplex = KEY1PLEX      Time = 10:13:13
                                CFRM Couple Data Set Information

Data Set Information
Volume Device FORMAT TOD          Data Set Name
-----  ---  -----  ---  -----
KEY1SP 260B 05/29/2004 08:51:30 SYS1.KEY1.PXESCDS
KEYUSR 261C 05/29/2004 08:47:42 SYS1.KEY1.AXESCDS

Control Card Information
MS POLICY CF STR CONNECT SMREBLD SMDUPLEX
--  -----  --  ---  -----  -----  -----
16   8   4   64      16      1      0

Policy Information
Cmds: D display policy / S start policy
Name      CF Str Date      Time      Userid
-----  --  ---  -----  -----  -----
_ BZOEPOL ACTIVE 2 19 02/10/2004 10:05:47 BZOE
_ HIRPOL   2 19 02/19/2004 19:45:57 HIR
_ HIRPOL1  1 8 05/25/2004 09:20:04 HIR

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
PF8=Forward  F9=Refresh     F12=Retrieve
  
```

Figure 79. CFRM Couple Data Set Information Panel before Policy Switch

The panel shows information about the names and locations of the CDSs. The panel also shows the parameters that were used by the formatting tool of XCF for the allocation of the CDS. The **POLICY** field, for example, displays the maximum number of policies the CDS can contain. Furthermore, the panel shows information about the policies in the CDS, for example, how many coupling facilities and structures are defined in every policy, and which policy is currently active.

To switch to the HIRPOL policy, enter S before this policy and press Enter. INGPLEX CDS displays the following confirmation panel:

```
INGKX30C          SA z/OS - Command Dialogs
Domain ID  = IPSFO  ----- INGPLEX CDS -----      Date = 03/01/01
Operator Id = NETOP1      Sysplex = KEY1PLEX          Time = 10:13:17

                          SETXCF START  Confirmation

You are going to start a new CFRM   CDS policy named  HIRPOL   .

The current policy

      BZOEPOL

will be stopped as soon as you proceed with the GO function key,

Command ==>
          F2=End      F3=Return      PF10=Go      F11=Cancel  F6=Roll
                                         F12=Retrieve
```

Figure 80. Confirmation Panel for Policy Switch

Displaying the Channel Paths for a CDS Type: In this example, the channel paths for the CFRM couple data sets are displayed.

Enter C before CFRM on the panel displayed in Figure 76 on page 159, and press Enter. The following panel is displayed:

How to Monitor and Control a Sysplex

```

INGKX318          SA z/OS - Command Dialogs          Line 1    of 4
Domain ID  = IPSFO          ----- INGPLEX CDS -----      Date = 03/02/01
Operator ID = NETOP1          Sysplex = KEY1PLEX          Time = 08:05:46
                                CFRM Channel Path Information
System   T  DEVN  CHPIDs                                     SSID
-----  -  ----  -----
KEY1     P  260A  E4=+ E5=+ E2=+ E3=+                               2600
         A  2610  E4=+ E5=+ E2=+ E3=+                               2600
KEY2     P  260A  E4=+ E5=+ E2=+ E3=+                               2600
         A  2610  E4=+ E5=+ E2=+ E3=+                               2600
KEY3     P  260A  E4=+ E5=+ E2=+ E3=+                               2600
         A  2610  E4=+ E5=+ E2=+ E3=+                               2600
KEY4     P  260A  13=+ 22=+ 30=+ 94=+                               2600
         A  2610  13=+ 22=+ 30=+ 94=+                               2600

Command ==>
F1=Help      F2=End          F3=Return      F6=Roll
              F9=Refresh      F12=Retrieve
  
```

Figure 81. Channel Path Information for CFRM Couple Data Sets

- The **System** column shows the name of the sysplex members.
- The **T** column (for 'type') indicates whether the CDS is the primary (value 'P') or alternate (value 'A').
- The **DEVN** displays the number of the device on which the CDS resides.
- The **CHPIDs** column shows the status of the paths to the devices in the format *chpid=status_code*. The codes are those of the operating system. They have the following meaning:
 - + The path is logically and physically available and I/O on the path was successful.
 - * The path is physically, but not logically available. The subchannel's logical path indicator is off but I/O to the path is successful. You can use the command `VARY PATH (ddd,nn),ONLINE` to make channel path *nn* logically available to device *ddd*.
 - The path is neither logically nor physically available. The subchannel's logical and physical indicators are both off for this channel path. You can use the command `CONFIG CHP(nn),ONLINE` to make the channel path logically available to all devices connected to the channel.
 - & The device is reserved to another path. This indicator applies to devices with the dynamic pathing selection feature.
 - < The path is installed but not physically available. The start subchannel request received a condition code of 3.
 - > The device microcode has detected an error and will not allow I/O to complete on the path.
 - B** The path is unable to communicate. The device indicates that a busy or reserve condition exists on the path.
 - C** A controller error occurred while accessing the device.
 - D** A device error occurred while accessing the device.

- I** Intervention is required; the device is not ready.
 - R** The path is available and the device is reserved to this path/group. This only applies to devices with the dynamic pathing feature.
 - T** A time out has occurred; there is no response from the device. The cause of the time out is undetermined and this condition is transient.
 - U** A storage control unit or storage director error occurred while accessing the device.
 - X** Unable to determine the failing unit.
- The SSID field displays the storage subsystem to which the device belongs.

INGPLEX IPL

With the INGPLEX IPL function you can record, view and compare the IPL information of the operating system. If a system does not behave after IPL as expected, the IPL recording function enables you to identify parameters that were changed, for example, since the last IPL. The recording function enables you to compare different IPL scenarios. INGPLEX IPL is a tool that helps to identify and resolve the cause of startup problems. The following information can be displayed:

- The selected system (or blank)
- The name of the sysplex
- The maximum number of IPLs that are stored for each system
- An indicator showing whether comments in PARMLIB members are ignored when collecting information

```

INGLX200          SA z/OS - Command Dialogs          Line 1 of 6
Domain ID = IPSFM  ----- INGPLEX IPL -----          Date = 02/22/04
Operator ID = NETOP1                               Time = 17:59:27

System . . . . . ==>          Max. number of IPL records/system : 10
Sysplex . . . . . ==> KEYIPLEX  Suppression of PARMLIB comments . : N
-----

Cmds: C compare record / D display details / E erase record

  System  IPL Timestamp      Dev  Volume  OpSys  Release  FMID
  -----  -----
-  KEYA    2004-02-22 13:52  770E 120204  z/OS  SP7.0.2  HBB7705
-  KEYA    2004-02-09 09:28  770E 120204  z/OS  SP7.0.2  HBB7705
-  KEYA    2004-02-08 15:28  770E 120204  z/OS  SP7.0.2  HBB7705
-  KEYA    2003-12-10 14:31  0707 120147  z/OS  SP7.0.2  HBB7705
-  KEYB    2004-02-22 13:59  770E 120204  z/OS  SP7.0.2  HBB7705
-  KEYB    2004-02-14 16:24  770E 120204  z/OS  SP7.0.2  HBB7705
-  KEYB    2004-02-11 18:46  770E 120204  z/OS  SP7.0.2  HBB7705
-  KEYB    2004-02-11 15:36  770E 120204  z/OS  SP7.0.2  HBB7705
-  KEYB    2004-02-11 14:22  770E 120204  z/OS  SP7.0.2  HBB7705

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
              F8=Forward  F9=Refresh  PF10=Previous F11=Next      F12=Retrieve
  
```

Figure 82. INGPLEX IPL Main Panel

Use PF10 and F11 to scroll through all available columns. SORT by column numbers is supported as well as the FIND and RFind command to locate information on the panel. You can also limit the display to a particular system by specifying the system name in the appropriate entry field.

The following command codes are available:

How to Monitor and Control a Sysplex

- C Compares the complete IPL information with another IPL record. A second panel will be displayed where you can select the second record.
- D Displays detailed information about this IPL record.
- E Erases the IPL information records. This action must be confirmed.
 - The **Sysplex** field shows the name of the sysplex.
 - The **System** column shows the name of the system in the sysplex.
 - The **IPL Timestamp** column shows the date and time of the IPL. The format is YYYY-MM-DD HH:MM converted to local time zone.
 - The **Dev** column shows the IPL device number.
 - The **Volume** column shows the volume serial of the IPL device.
 - The **OpSys** column shows the name of the operating system, for example, z/OS or OS/390.
 - The **Release** column shows the release level of the operating system.
 - The **FMID** column shows the FMID of the operating system.

For further information about the panel fields refer to the online help.

INGPLEX DUMP

The dump functions can be invoked directly by specifying the commands, or from the dump panel of the INGPLEX command selecting the appropriate command. In addition, you can invoke the dump submenu from the main panel of the INGPLEX command selecting command 7. The following panel is displayed:

```
INGLX250          SA z/OS - Command Dialogs          Line 1 of 12
Domain Id = IPSFP  ----- INGPLEX -----          Date = 02/26/04
Operator Id = NETOP1                               Time = 16:30:36

Sysplex . . . . . : KEY1PLEX

Select the desired command:                          INGPLEX ...

  1 Control default SDUMP options                     SDUMP
  2 Issue SVC dumps                                   SVCDUMP
  3 Control SLIP trap settings                        SLIP

Command ==>
F1=Help      F2=End      F3=Return
F6=Roll      F12=Retrieve
```

Figure 83. INGPLEX DUMP Panel

INGPLEX SDUMP: If you select option 1, the following panel is displayed:

```

INGLX251          SA z/OS - Command Dialogs          Line 1 of 12
Domain Id = IPSFP  ----- INGPLEX SDUMP -----      Date = 02/26/04
Operator Id = NETOP1                               Time = 15:44:58

Sysplex . . . . ==> KEY1PLEX          Permission . . . . : ALL
-----
Cmds: C change

  System      Dump options
-----
_ KEY1      Q=      Type=      Buffers=  0K  MaxSpace=  500M  MsgTime=99999
                                     LSQA
                                     TRT
_ KEY2      Q=      Type=      Buffers=  0K  MaxSpace=  500M  MsgTime=99999
                                     LSQA
                                     TRT
_ KEY3      Q=      Type=      Buffers=  0K  MaxSpace=  500M  MsgTime=99999
                                     LSQA
                                     TRT
_ KEY4      Q=      Type=      Buffers=  0K  MaxSpace=  500M  MsgTime=99999

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
              F8=Forward  F9=Refresh      F12=Retrieve
  
```

Figure 84. INGPLEX SDUMP Panel

The following command code is available:

C change

Invokes the modification panel by providing the options of the selected system as input

- The **Sysplex** field shows the name of the sysplex.
- The **System** field shows the name of the system in the sysplex.
- The **Permission** field shows your authorization level.
- The **Dump options** field shows the default SDUMP options of all systems in the sysplex. For each system the following details are displayed:

Q= Shows whether or not SDUMP quiesces the system while dumping the contents of the SQA or CSA.

TYPE=

Causes SVC dump to dump the cross memory address spaces that the caller has when SVC dump gets control (XMEM) or when the error causing the dump occurs (XMEME).

BUFFERS=

Shows the reserved storage exclusively used by SVC dump. This storage can be used while capturing the contents of the common area storage.

MaxSpace

Shows the maximum amount of virtual storage that SVC dump can use to capture volatile virtual storage data, summary dump data, and component-specific data before writing the dump to DASD.

MsgTime

Shows for which amount of time (mm) the message IEA793A is shown at the console. When the system deletes the message, it also deletes the captured dump.

The FIND and RFIND commands are supported. If you specify command code C, the following panel is displayed:

How to Monitor and Control a Sysplex

```

INGLX252          SA z/OS - Command Dialogs          Line 1 of 12
Domain Id = IPSFP ----- INGPLEX SDUMP -----      Date = 02/26/04
Operator Id = NETOP1                                   Time = 16:18:08

System . . . . . : KEY1
Sysplex . . . . . : KEY1PLEX          Recommended options are underlined.

NODUMP ... ==> N      (all other options below are ignored)

ALLNUC ... ==> ALLPSA(*) ==> COUPLE ... ==> CSA ..... ==> NUC ..... ==> GRSQ ..... ==> LPA ..... ==> LSQA ..... ==> Y
PSA ..... ==> RGN ..... ==> SERVERS .. ==> SQA(*) ... ==> SUMSUMP(*) ==> SWA ..... ==> TRT ..... ==> Y
XESDATA .. ==> ---      (*) = The NOxxx option is generated when not selected.

Q(uiesce) ==> ---      (YES / NO)
Type ..... ==> ---      (XMEM / XMEME)
Buffers .. ==> 0K       (nnnnK / nnnM)
MaxSpace . ==> 500      (MB)
MsgTime .. ==> 99999   (minutes)

Command ==>
F1=Help      F2=End      F3=Return    F4=Set SYS  F5=Undo all  F6=Roll
PF10=Set SYSS F11=Set SYSP F12=Retrieve

```

Figure 85. INGPLEX SDUMP Modification Panel

The modification panel allows you to modify all SDUMP options. Furthermore, you can delete SDUMP options. After entering your changes you can set the new options for:

- The selected system
- All systems in the sysplex
- Selected systems in the sysplex

To set the options press the appropriate F-key. If you want to modify selected systems in the sysplex, you are prompted for the systems on which the SDUMP options are being changed. To reset the options to the state when the modification panel was invoked press F5 Undo all.

Note: The user must be authorized to change any SDUMP option. The authorization can be any of those which are used for controlling coupling facilities and couple data sets.

For further information about the panel fields refer to the online help.

INGPLEX SVCDUMP: With the INGPLEX SVCDUMP command you can display the dump status information of each system in the sysplex. The INGPLEX SVCDUMP function allows you to issue a multisystem dump of up to 15 address spaces of a single system including their data spaces and structures.

```

INGLX26S          SA z/OS - Command Dialogs          Line 1 of 6
Domain Id = IPSFP  ----- INGPLEX SVCDDUMP -----   Date = 02/06/04
Operator Id = NETOP1                                Time = 17:05:17

The following systems of sysplex KEY1PLEX are registered to the automation.
Use any non-blank character to select one system and then press ENTER.

      Sel  System
      ---  -----
      -    KEY2
      -    KEY3
      -    KEY4

Command ==>
      F1=Help      F2=End      F3=Return      F6=Roll
                  F12=Retrieve
    
```

Figure 86. INGPLEX SVCDDUMP Target System Selection Panel

- The **Sel** field lets you select a system for which a dump should be taken.
- The **System** field shows the name of the system having joined the same XCF group the operator is logged on to.

For further information about the panel fields refer to the online help. After selecting a system and pressing Enter, the following panel is displayed:

```

INGLX260          SA z/OS - Command Dialogs          Line 38 of 63
Domain Id = IPXFG  ----- INGPLEX SVCDDUMP -----   Date = 03/11/04
Operator Id = NETOP1                                Time = 12:26:26

System . . . . . ==> KEYA
Sysplex . . . . . ==> KEYAPLEX
-----
Cmds: D/S de-/select job names for the SVC dump (up to 15 can be specified)

      Jobname  ASID  WorkUnitID  Userid
      -----  ----  -
      - TNF      0024
      - TRACE   0004
      - TSO      003B  STC05983   STCUSER
      - VLF      0019
      - VMCF     0025
      - VTAM     001E  STC05982   STCUSER   selected
      - WATS     0217  TSU06587   -         selected
      - WLM      000B
      - XCFAS    0006

Command ==>
      F1=Help      F2=End      F3=Return      F5=NextPnl      F6=Roll
                  F8=Forward    F9=Refresh
    
```

Figure 87. INGPLEX SVCDDUMP Address Space Selection Panel

How to Monitor and Control a Sysplex

If you select VTAM address space and WATS address space, which is a user, press Enter and then PF5, the following panel is displayed:

```
INGLX261          SA z/OS - Command Dialogs          Line 1 of 9
Domain Id = IPXFG  ----- INGPLEX SVCDUMP -----      Date = 03/11/04
Operator Id = NETOP1                                     Time = 12:34:04

System . . . . . : KEYA
Sysplex . . . . . : KEYAPLEX
-----
Cmds: D/S de-/select the areas to be dumped (max. 113 structures)
      A selection of the job name includes all related areas.

  Jobname  ASID  T Data Space/XCF Group Member/Structure
  -----  ---  - -----
  VTAM     001E
  -                D IST90C95
  -                D 00012IXL
  -                D 00013IXL
  -                L ISTGENERIC
  -                M ISTCFS01.KEYB.VTAM.IPXVH__DEIBMIPS      selected
  -                M ISTXCF.KEYB.VTAM.IPXVH__DEIBMIPS      selected
  -                M IXCLO008.KEYB.VTAM.M28                 selected
  WATS     0217
Command ==>
F1=Help      F2=End      F3=Return  F4=PrevPnl  F5=NextPnl  F6=Roll
              F8=Forward  F9=Refresh                F12=Retrieve
```

Figure 88. INGPLEX SVCDUMP Address Space Detail Panel

Address space VTAM has many data spaces (D), one list structure (L) and some XCF group members (M). TSO user WATS has nothing.

The following command codes are supported:

- D** Deselects the previous selection
- S** Selects a local address space, data space, structure, or XCF group member address space for the SVC dump.

If you press PF5, the dump option selection panel is displayed:

```

INGLX262                SA z/OS - Command Dialogs
Domain Id = IPSFP      ----- INGPLEX SVC -----          Date = 02/26/04
Operator Id = NETOP1                                     Time = 18:02:56

System . . . . . : KEY3
Sysplex . . . . . : KEY1PLEX

Title .... ==> _____
                ==> _____

SDATA Dump Options (recommended options are underlined)
ALLNUC ... ==> -   ALLPSA(*) ==> Y   COUPLE ... ==> -   CSA ..... ==> Y
GRSQ ..... ==> Y   LPA ..... ==> -   LSQA ..... ==> -   NUC ..... ==> Y
PSA ..... ==> -   RGN ..... ==> Y   SERVERS .. ==> -   SQA(*) ... ==> Y
SUMSUMP(*) ==> Y   SWA ..... ==> -   TRT ..... ==> Y   WLM ..... ==> -
XESDATA .. ==> -   (*) = The NOxxx option is used when not selected.

Structure Dump Options (SUMMARY and ADJUNCT/ENTRYDATA are mutually exclusive)
COCLASS .. ==> -   EMCONTROLS ==> -   LISTNUM .. ==> -   STGCLASS . ==> -
ADJUNCT .. ==> -   ENTRYDATA  ==> -   SUMMARY .. ==> -

Command ==>>>
F1=Help      F2=End      F3=Return    F4=PrevPn1  F5=Dump      F6=Roll
                                           F12=Retrieve
    
```

Figure 89. INGPLEX SVCDUMP Dump Option Panel

The panel shows the default dump options being set on invocation. After specifying the dump title, press F5 to issue the dump. When the dump is taken, the function returns to the address space selection panel with all selections cleared. The SORT, FIND and RFIND commands are supported for selection panels only. For further information about the panel fields refer to the online help.

INGPLEX SLIP: With the INGPLEX SLIP command you can display serviceability level indication processing (SLIP) traps being set at all systems in the sysplex. With INGPLEX SLIP you can view, enable, disable, and delete the SLIP trap defined in the sysplex.

How to Monitor and Control a Sysplex

```
INGLX270          SA z/OS - Command Dialogs          Line 1 of 96
Domain Id = IPSFP ----- INGPLEX SLIP -----      Date = 02/26/04
Operator Id = NETOP1                                Time = 18:20:21

System . . . . . ==> _____ (leave blank for all systems)
Slip Trap Id . . ==> _____ (leave blank for all ids)
Sysplex . . . . . ==> KEYIPLEX      Permission . . . . . : ALL
-----

Cmds: +/- display/hide settings D disable E enable R remove

  System   Id     Status   Settings
  -----
- KEY1     XB37   ENABLED
- KEY1     XD37   ENABLED
- KEY1     XE37   ENABLED
- KEY1     X0E7   ENABLED
- KEY1     X0F3   ENABLED
- KEY1     X013   ENABLED
- KEY1     X028   ENABLED
- KEY1     X13E   ENABLED

Command ==>
F1=Help      F2=End      F3=Return   F6=Roll
              F8=Forward  F9=Refresh  F12=Retrieve
```

Figure 90. INGPLEX SLIP Main Panel

The following command codes are available:

- + Shows the settings of the SLIP trap.
- Hides the settings of the SLIP trap.
- D Disables the SLIP trap.
- E Enables the SLIP trap.
- R Deletes the SLIP trap.

The SORT, FIND and RFIND commands are supported.

Note: The user must be authorized to enable, disable, and delete a SLIP trap. The authorization can be any of those which are used for controlling coupling facilities and couple data sets.

For information about the panel fields refer to the online help.

INGPLEX BESTpractices

This command allows you to view the currently active best practices from the system doing global checks, which is shown on the panel.

When you issue the INGPLEX BEST command, to display the IBM and User-defined best practices, the panel in Figure 91 on page 173 is displayed.


```

INGLX351          SA z/OS - Command Dialogs          Line 139 of 217
Domain ID = IPXNH      -- INGPLEX BESTPRACTICES --      Date = 04/10/04
Operator ID = KHH                               Time = 09:50:09

System . . . . . ==> KEYB                      Sysplex . . . . . : KEYAPLEX
-----
Check
  Keyword      IBM policy value                      D User policy value
-----
AVAILABLE_FRAME_QUEUE_THRESHOLDS
Severity..: High
Parms.....: 400,600,200,400                        * 401,601,201,401
Interval...: 24:00                                  * 0:10
Version...: HBB7703
Date.....: 20030211                                * 20030410
Reason....: System may not recover in time         * WE NEED OTHER PARAMETERS FOR
              if set too low                       * OUR SYSTEM
REAL_STORAGE_AVAILABILITY
Severity..: Low
Interval...: 24:00

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
F7=Backward  F8=Forward  F9=Refresh    F12=Retrieve
  
```

Figure 91. INGPLEX BESTpractices Panel

This panel displays the IBM's and user best practices of the HealthChecker function. The following fields are shown in the nonscrollable area of the screen.

- System** Name of the system from which the best practices are retrieved.
- The best practices can only be retrieved from a system where the HealthChecker function is active. If multiple systems are eligible, the system where the checks with sysplex scope (global checks) are done is chosen by default.
- You may choose another system by overtyping the system name with the name of the desired system. If the system name is cleared, the best practices are retrieved from the default system.
- Sysplex** Name of the sysplex that the system named above is part of.

The scrollable area of the screen lists the IBM best practice recommendation and, if defined, the user override value.

- Check** Best practice checkname is displayed in white.
- Keyword** Associated keywords for a check are displayed below the checkname.
- IBM policy value** Details the IBM Bestpractice value. The following values are defined:
- Severity** Expresses a sense of urgency about the need to fix the exceptional situation for the check. Possible values are High, Medium, Low.
 - Parms** IBM's recommended settings for the check.
 - Interval** IBM's recommendation of how often to perform the check. The time interval is shown in the format *hh:mm*.

How to Monitor and Control a Sysplex

Version	Shows the FMID of the operating system release for which the check is applicable. This can be a range.
Date	Shows the date when IBM introduced or last modified the check.
Reason	Documents the reason why the check should be done.

D An asterisk in this column represents a discrepancy between the IBM recommended value and a User override.

User policy value

Details the user override values. The following values can be overridden:

- Parms, where applicable
- Interval
- Date
- Reason
- Severity

The value for Interval is also used to mark checks which are not performed on the system for the following reasons:

NOCALL specified

Indicates that in the user policy the check is specified explicitly with NOCALL.

NOCALL - Check failed

Indicates the checker failed 3 times.

NOCALL - System error

Indicates an unrecoverable system error in this check. It might also indicate an error in the IBM Parms.

NOCALL - User Error

Indicates an error in the user parameters for this check.

NOCALL - n/a

Indicates the check is not applicable on the system. For example if the system is set up in Monoplex mode, certain checks are not applicable.

NOCALL - Global check n/a

Indicates that the check is not executed because some other system performs the global checks.

NOCALL - ?????

Indicates that the check is not executed for some undefined reason.

When a User value does not match an IBM recommended value, along with the discrepancy flag, the user value is displayed in a different color.

The display of a particular keyword is suppressed if it is not applicable to the check.

How to Work with INGHC

How to Work with INGHC

Purpose: This command serves two purposes:

1. Allows you to view the reports of the IBM Health Checker for z/OS and Sysplex (HealthChecker)
2. Allows you to request the HealthChecker to perform individual checks according to your filtering options

Note: If you want to override IBM's best practice values with your own values, use the INGAMS REFRESH command.

Types of Reports:

About the Reports: The following rule applies for each check overridden in the user policy: If there is some error in your specification, the check is not performed at all, that is, the IBM values are not used either. To make the check active again, remove your override or fix it.

The HealthChecker reports reflect values at a point in time (snapshot). The report is comprised of a series of records in the System Logger. These records have the following components:

- Message text and explanation
- Recommendations of actions that can be taken to address an exception
- IBM suggestions
- Reasons for IBM's suggestions

The HealthChecker generates two report formats:

- **Regular**

The regular format produces confirmation messages of those checks where the results meet either IBM's or your override values. For selected checks, such as several related to consoles or UNIX[®] System Services file systems, additional information about these resources is provided.

- **Exception**

The exception format provides status on checks that do not meet the criteria for the check.

To help distinguish a successful check from one encountering an exception, the record is explicitly marked in column 'E' as an I(BM) or U(ser) exception. In addition, the status of an exception is explicitly noted in the reports as:

Exception: IBM criteria not met

or

Exception: User criteria not met

Unless you take any of the actions given in "Types of Actions," the HealthChecker performs regular checks at predefined time intervals. The time intervals are defined individually for each check as part of IBM's best practices and can be overridden. The checks are done based on IBM's best practices or your overrides.

Types of Actions: You can request the HealthChecker to run selected checks or all checks at a point in time when you want it to.

Recommendation: You should iteratively run the HealthChecker function and take corrective action until you have no exceptions. You should either update the values used in your environment, override the IBM values, or suppress the IBM check. This is important so that on subsequent runs of the HealthChecker, you will only see exceptions that you should attend to. Otherwise, the reports may contain a mixture of messages that you regularly ignore and those that could reflect a new potential problem, making it more likely that you could miss a key exception message.

Minimally, you should run the HealthChecker when you reIPL. However, many values can change between IPLs. You may therefore want to consider the following:

- permanently having the HealthChecker perform the checks repeatedly based on their respective, defined time interval. If the time intervals that are predefined by IBM do not suit your needs, you can always override them for individual checks using the TIMEINT parameter.
- if you don't want to have the HealthChecker permanently active, you can activate it at regular intervals (for example, once a day) and then deactivate it.

Examples: When you issue the INGHHC command, the panel in Figure 92 is displayed.

```

INGLX350                SA z/OS - Command Dialogs                Line 1    of 82
Domain Id = IPXNH      ----- INGHC -----                    Date = 04/10/03
Operator Id = KHH      Sysplex = KEYAPLEX                        Time = 09:53:16

Start time . ==> 2003-04-10 08:53:00    format: yyyy-mm-dd hh:mm:ss
End time ... ==>                               Exceptions . ==> Latest Check ==>
Consoles ... ==> X CDS ..... ==> X CF/STR ..... ==> X XCF signal. ==> X
XCF recovery ==> X System Res. ==> X Other ..... ==> X
System(s) .. ==>

-----
Cmds: + / - Expand/Collapse check results

Timestamp            System  E S Footprint
-----
2003-04-10 09:52:15 KEYA  I H USS_FILESYS_CONFIG
- 2003-04-10 09:52:15 KEYA  M SDUMP_AVAILABILITY
- 2003-04-10 09:52:15 KEYA  I M XCF_FAILURE_DETECTION_INTERVAL
2003-04-10 09:52:15 KEYA  I L XCF_CLEANUP_VALUE
- 2003-04-10 09:52:15 KEYA  L RSU_STORAGE_AVAILABILITY
- 2003-04-10 09:52:15 KEYA  I L REAL_STORAGE_AVAILABILITY
Command ==>
F1=Help      F2=End      F3=Return      F5=Coll. All  F6=Roll
              F8=Forward  F9=Refresh    F10=NewCheck  F12=Retrieve
    
```

Figure 92. INGHHC Panel

This panel displays the results of all Health Check requests. The fields in the non-scrollable area, in the top portion of the screen, allow filtering of Health Check requests as follows:

- Start time** The date and time start filter in YYYY-MM-DD HH:MM:SS format converted to the local timezone. This defaults to the current time less one hour if not provided.
- End time** The date and time end filter in YYYY-MM-DD HH:MM:SS format converted to the local timezone. This defaults to the current time if not provided.

- Exceptions** The Exceptions filter returns *only* checks that show an exception that occurred between the Start and End times. This display can also be limited by using the other filters in this area.
- Latest Check** The Latestcheck filter returns *only* the latest checks that have been performed. This display can also be limited by using the other filters in this area. You may have no output if the Latestcheck was performed at a time outside the Start and End times provided.
- System(s)** The System(s) field enables you to restrict data to a specific system or group of systems in a sysplex.

Note: All systems are assumed if this field is left blank.

The following allows you to filter the checks you are interested in. You cannot filter individual checks, however, because the checks are grouped into categories. The following categories have been defined:

- Consoles** The Consoles field returns all console-related results.
- CDS** The CDS field returns all Couple-Data-Set-related results.
- CF/STR** The CF/STR field returns all Coupling-Facility-Structure-related results.
- XCF signal.** The XCF Signal field returns all XCF-Signalling-related results.
- XCF recovery** The XCF Recovery field returns all XCF-RECOVERY-related results.
- System Res.** The System Res. field returns all system-related check results.
- Other** The Other field returns all data not covered in the previous filters.

The lower portion of the panel is a scrollable area where the results of your requests are displayed. The output fields contain the data in the following format:

- Timestamp** The Timestamp field indicates the actual date and time that the health check was performed.
- System** The System field displays the system that the health check was performed on.
- E(xception)** The Exception field indicates whether there is a deviation from the specified best practices (either IBM or User defined). An 'I' will indicate an IBM exception exists, a 'U' indicates that a User exception exists. A blank indicates that there are no exceptions.
- S(everity)** The Severity field indicates the importance of the item. Its value is defined in the IBM best practices or user overrides. This field contains either L (Low), M (Medium), H (High).
- Footprint** The Footprint contains the name of the check item. For the list of all valid names see the online help. To view an item or group of items in more detail, enter a '+' in the field to the left of the item(s) you wish to display and press Enter. To reset the display, enter a '-' in the field to the left of the item and press Enter to collapse it. The amount of detail displayed is limited by the size of one record in the system logger. If the output exceeds this limit, it is truncated, and the respective footprint is displayed in a different color.

Two PF keys are specific to this panel.

Coll. All (PF5) Collapses the detail description of all check results.

NewCheck (PF10)

Requests the HealthChecker to immediately run checks as specified in the filter options. The request is sent to all specified systems, the default is ALL. The report data is displayed, according to the filter options.

Note: Even if the input fields of the filter options show only an 'X' when selected they must be 2 characters long to support up to 26 footprint lines due to NetView restrictions.

Note: You should check that the systems listed in the New Check confirmation panel are those for which you want to perform a new check. This is because PF keys work such that they do not accept new values in input fields. Instead, they reset the input fields to the values last entered with the Enter key and then run the requested function. If Newcheck is wanted for another system or for some other filter in general (systems are a filter criterion) you first need to change those values and press Enter. The new values are then validated for correctness, and then the respective PF key can be executed correctly.

Pressing the PF10 key to request a NewCheck displays the Health Check confirmation screen, as shown in Figure 93.

```

INGLX35C          SA z/OS - Command Dialogs
Domain ID = IPSFM  ----- INGHC -----      Date = 07/20/02
Operator ID = HIR          Sysplex = KEY1PLEX      Time = 17:13:51

                H E A L T H   C H E C K Confirmation

You have requested an additional check of the resources below on the indicated
systems. Press the GO function to perform the request.

Check(s) . .: CONSOLES      CDS          CF/STR      XCF SIGNAL.
              XCF RECOVERY  SYSTEM RES.  OTHER

System(s)  .: *ALL

Command ==>
              F2=End          F3=Return          F6=Roll
                          F10=Go          F11=Cancel          F12=Retrieve

```

Figure 93. INGH C Check Confirmation Panel

This panel shows the selected check criteria to be performed and the systems involved. If you want to cancel the new check, press PF11, otherwise press PF10 to perform the request.

The output, when returned, will be sorted in descending Timestamp order, as shown in Figure 92 on page 176. You may sort the output by issuing the SORT command on the command line, followed by the sort direction (ascending or descending) and the column to sort. For example, to sort on Severity in descending order issue:

sort d 4

To search for text strings, issue the FIND, F, RFIND or RF commands. For example, to find the string *consoles*, enter the following:

```
f consoles n
```

Note: The FIND command can search *only* the text for those check results that have been expanded, even when the text is not visible on the action screen.

For more information about sorting output and searching for text strings, refer to the online help.

To view an item or group of items in more detail, enter a '+' in the field to the left of the item(s) you wish to display. To reset the display, enter a '-' in the field to the left of the item to 'collapse', or use PF5 to collapse all expanded items.

Line Mode Output: If you issue the INGHG command with OUTMODE=LINE, the output shows both the footprint data as well as the detail data, and will be similar to that shown in Figure 94.

```

NVSS V 1 R 4 IPXNG      Tivoli NetView  IPXNG XJIVENS  03/11/03 12:30:06
| IPXNG
Sysplex = KEYAPLEX

Timestamp              System    E S Footprint              11 Mar 2003 12:29:05
-----
2003-03-11 12:10:50 KEYB      HEALTH_CHECKER_ENDED
      z/OS Sysplex Health-Checker Version 01.02 5 Mar 2003 06:53:
      Ended for system KEYB in sysplex KEYAPLEX
      At 12:10:50 on 11 Mar 2003
2003-03-11 12:10:50 KEYB      I H CDS_DATASET_SEPARATION
      The following PRIMARY couple datasets resides on Unique volumes:
      SYS1.KEYAPLEX.PXCFCDs: Primary SYSPLEX Couple dataset

      This is consistent with the IBM recommendation that the primary
      SYSPLEX Couple dataset, the primary CFRM Couple dataset, and the
      primary LOGR Couple dataset, should be placed on different volumes.

      CDS_DATASET_SEPARATION      *Exception: IBM Criteria not met*
      The following PRIMARY couple datasets resides on the same volume
      (KEYAXP):
=χ= ***

```

Figure 94. INGHG Line Mode Output

How SA z/OS Uses Error Thresholds

Error thresholds influence whether SA z/OS recovers from an error situation or not. For applications, you can define a critical threshold for restarting these applications. This is a number of error conditions within a certain time interval, for example, five error conditions requiring restart within one hour. During a condition requiring restart, SA z/OS checks whether the number of occurrences of the condition reaches the critical threshold. If it is reached, SA z/OS does not attempt to restart a resource.

For z/OS components, such as dump data sets or log data sets, you can define thresholds to limit the frequency of how often they may be deleted after they have filled up without an action being taken or a notification being sent to the operator.

How SA z/OS Uses Error Thresholds

Error thresholds also determine when you should be alerted to problems. The primary use of error thresholds is to track subsystem abends and ensure that the abend and restart cycle does not become an infinite loop, but they may also be customized for other uses.

Refer to *System Automation for z/OS Defining Automation Policy* for information on how to define error thresholds. The following sections describe how to obtain information about them.

Displaying Threshold Settings and Occurrences

SA z/OS counts the number of times that a subsystem abends. Each time, an abending subsystem is restarted, a check is made to see if the restart is done too often. SA z/OS allows you to specify thresholds in terms of error rates. If the measured error rate reaches the *critical* threshold, SA z/OS stops to recover the resource and posts it to a broken status.

You have the following possibilities to both obtain information about threshold settings and to view when thresholds have been reached:

INGTHRES

If you want to display all defined thresholds or add, change, or delete threshold settings for a particular resource, use the INGTHRES command. This displays the related *Command Dialogs* panel similar to the one shown in Figure 95 on page 181.

DISPERRS

If you want to check the error counts for all resources on a specific system, you should issue a DISPERRS command to this target system. SA z/OS will display the DISPERRS *Command Dialogs* panel with a list of all the errors that have been recorded for all resources on the target system (see “How to Display Occurred Errors - DISPERRS” on page 181).

DISPASF

If you want to check the error counts for a specific resource on a specific target system, issue the DISPASF command.

How to Display Threshold Settings for a Selected Resource - INGTHRES

Type INGTHRES on any command line to display a panel similar to the following that shows the settings for the critical, frequent, and infrequent thresholds for specified resource:


```

INGKYTH0          SA z/OS - Command Dialogs          Line 1 of 77
Domain ID = IPSFM ----- INGTHRES -----          Date = 04/12/04
Operator ID = OPER1                                     Time = 17:33:52

Cmd:  A Add thresholds   C Change thresholds   D Delete thresholds

Cmd  System  Resource          Critical    Frequent    Infrequent
-----
KEY1  CICS1G.SOS  4 in 00:05    4 in 01:00    7 in 24:00
KEY1  CICS1G.TRAN 2 in 00:05    4 in 01:00    5 in 24:00
KEY1  CICS1G.VIOL 3 in 00:05    3 in 01:00    6 in 24:00
KEY1  CICS1H.SOS  4 in 00:05    4 in 01:00    7 in 24:00
KEY1  CICS1H.TRAN 2 in 00:05    4 in 01:00    5 in 24:00
KEY1  CICS1H.VIOL 3 in 00:05    3 in 01:00    6 in 24:00
KEY1  CICS3E.SOS  4 in 00:05    4 in 01:00    7 in 24:00
KEY1  CICS3E.TRAN 2 in 00:05    4 in 01:00    5 in 24:00
KEY1  CICS3E.VIOL 3 in 00:05    3 in 01:00    6 in 24:00
KEY1  DB2.DSNL008I 5 in 01:01    3 in 02:02    1 in 03:03
KEY1  DB2C.DSNL008I 5 in 02:00    3 in 03:00    2 in 05:00
KEY1  DB2T        5 in 00:30    3 in 00:30    2 in 00:30

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Roll
               PF8=Forward  PF9=Refresh  PF12=Retrieve
    
```

Figure 95. INGTHRES Command Dialog

How to Display Occurred Errors - DISPERRS

If you specify the DISPERRS command, you get the DISPERRS command dialog, which displays the number of errors for all the applications on the target system and also when an application has reached its threshold. Scroll through the panel to find the resource that has failed.

How SA z/OS Uses Error Thresholds

```

AOFKAAD5          SA z/OS - Command Dialogs          Line 1 of 19
Domain ID = IPUFM  ----- DISPERRS -----          Date = 10/28/03
Operator ID = OPER1                                Time = 15:30:24

  System  Resource  Type      Thrs  No Date  Time
-----
AOC7     AOFIPLDT  CONTROL              10 03/03/04 12:11
AOC7                                     9 03/03/04 09:22
AOC7                                     8 03/01/04 12:12
AOC7                                     7 02/24/04 10:32
AOC7                                     6 02/18/04 08:13
AOC7                                     5 02/05/04 14:26
AOC7                                     4 02/01/04 15:42
AOC7                                     3 01/25/04 07:49
AOC7                                     2 01/15/04 12:01
AOC7                                     1 01/04/04 07:21
AOC7     THRY000NN10  SUBSYSTEM  CRIT  3 02/25/04 13:15
AOC7                                     2 02/25/04 13:14
AOC7                                     1 02/25/04 13:00
AOC7     TRGG007AA00  SUBSYSTEM  CRIT  1 02/19/04 09:38
AOC7     TRGG007AA10  SUBSYSTEM  CRIT  1 02/19/04 09:38
AOC7     MVSDUMP      MVSESA     CRIT 10 01/25/04 08:23
AOC7                                     9 01/25/04 08:08
AOC7                                     8 01/21/04 07:21
AOC7                                     7 01/21/04 07:17
AOC7                                     6 12/15/03 23:31
AOC7                                     5 12/15/03 23:31
AOC7                                     4 12/15/03 23:30

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Roll
               PF9=Refresh   PF12=Retrieve

```

Figure 96. DISPERRS Command Dialogs Panel

In this sample screen, the AOFIPLDT entry shows when the last IPL was performed on the system. The THRY000NN10 subsystem is shown as having reached its *critical* threshold at 13:15 on 02/25/99.

If a subsystem is in the BROKEN status and it has not exceeded its critical threshold, then it has probably encountered an unrecoverable error. If you look into either SDF or DISPINFO for it, you should find the text of the AOF571I status change notification message, which will explain why it was posted to BROKEN.

How to Display Occurred Errors for a Specific Resource - DISPASF

While DISPERRS displays occurred errors for all a system's resources, DISPASF lets you see detailed information about errors for a resource on a specific system. For example, if you want to see more information about the error with resource *MVSDUMP* on system *AOC7* from the panel in Figure 96, issue the following command:

```
DISPASF MVSDUMP TARGET=AOC7
```

to display the following information:

```

AOFK3D0X          SA z/OS - Command Response          Line 1 of 19
Domain ID = IPUFM ----- DISPASF -----          Date = 03/04/04
Operator ID = OPER1                                     Time = 12:35:42

Command = ASF ID=MVSDUMP,REQ=DISP
SYSTEM = AOC7          STATISTICS DISPLAY REQUESTED FOR MVSDUMP
-----
STATISTICS DISPLAY REQUESTED FOR MVSDUMP
ID= MVSDUMP          , TYPE= MVSESA          , STATUS= DOWN
LAST UPDATE BY OPERATOR AUTREC
LAST THRESHOLD EXCEEDED - CRIT
OPERATOR NOTIFIED: N
LAST STATUS CHANGE DATE= 08/08/03 , TIME= 06:26 , OPID= AUTREC
LAST MONITORED DATE= 08/08/03 , TIME= 06:26
  ERROR COUNT      DATE      TIME
      01      12/03/03      21:56
      02      12/03/03      21:56
      03      12/15/03      23:30
      04      12/15/03      23:30
      05      12/15/03      23:31
      06      12/15/03      23:31
      07      01/21/04      07:17
      08      01/21/04      07:21
      09      01/25/04      08:08
      10      01/25/04      08:23
END OF MULTI-LINE MESSAGE GROUP

Command ==>
PF1=Help      PF2=End      PF3=Return      PF6=Ro11
                PF9=Refresh      PF12=Retrieve
    
```

Figure 97. DISPASF Command Response

How to Set or Change Error Thresholds

You can set or change thresholds for a resource, the resource group defaults, or the system defaults using the INGTHRES command dialog.

1. Follow the instructions in “How to Display Threshold Settings for a Selected Resource - INGTHRES” on page 180 to specify the resource for which you want to set thresholds and to access the INGTHRES command dialog.
2. To change the Critical, Frequent, or Infrequent thresholds, type the number of errors and specify the time period.

Handling Jobs that are not Controlled by SA z/OS

The INGLKUP command dialog displays all active jobs, started tasks (STC), and APPC/MVS transaction programs that are not controlled by SA z/OS. You can either stop, cancel, or force these jobs.

Handling Jobs that are not Controlled by SA z/OS

```
INGKYL00          SA z/OS - Command Dialogs          Line 1 of 23
Domain ID = IPSFM ----- INGLKUP -----          Date = 04/12/04
Operator ID = OPER1          System = KEY1          Time = 17:43:59

Cmd: C cancel job          F force job          P stop job

Cmd Job          Step          Procedure Type ASID
-----
      AMOY          AMOY          AMPROC          SO
      AMOZ          AMOZ          AMPROC          SO
      AVM           AVM           AVM             S
      ENQMON        ENQMON        ENQMON          S
C   FTDP1          STEP1         FTPOE           AO          ** CANCELLED **
      FTPOE1        STEP1         FTPOE           AO
      IMS711I1      IMS711I1     IMS711I1        S
      IMS711PP      IMS711PP     IMS711PP        S
      IMS712I1      IMS712I1     IMS712I1        S
      IMS713I1      IMS713I1     IMS713I1        S
      IMS811I1      IMS811I1     IMS811I1        S

Command ==>
PF1=Help          PF2=End          PF3=Return          PF6=Ro11
                  PF8=Forward        PF9=Refresh        PF12=Retrieve
```

Figure 98. INGLKUP Command Dialog Panel

For each job the following information is displayed:

- job name
- step name
- procedure name
- job type
- address space ID (this is only shown when the job name is not unique)

Communicating with Other Systems

This chapter discusses the following topics:

- “Sending Commands within a Sysplex”
- “Terminal Access Facility Full-Screen Sessions” on page 188
- “Gateway Sessions” on page 190

Sending Commands within a Sysplex

This section informs you about the prerequisites for communicating with systems

- within a sysplex; this scenario is called *single system image*.
- within your complete enterprise; this scenario is called *single point of control for the enterprise*.

There is also a series of commands that are directly processed by the automation manager of a sysplex. For these commands, you need no target parameter. You can retrieve sysplexwide information, such as the statuses of resources within the sysplex. For a complete list of sysplexwide commands, refer to the *System Automation for z/OS Operator's Commands*.

You can use a system within the sysplex to monitor and manipulate all other systems in the sysplex. Then each system in the sysplex looks like the one you are currently working on (single system image).

If your system is the focal point system, you can use it to manipulate all resources of your enterprise from this focal point system issuing NetView commands. The target systems need not be members of your sysplex (single point of control).

To exploit these features, the following prerequisites and conditions need to be fulfilled:

- The target system is within the local sysplex (this is valid for single system image) or the target system has a direct gateway connection with the local system (this is valid for single point of control).

The following prerequisites need to be fulfilled for both cases:

- The target has SA OS/390 1.3 or higher installed.
- The target is active.
- SA z/OS on the target has been fully initialized.
- The target can be reached either by XCF or RMTCMD.

SA z/OS will search your environment for the target system in the following order:

- system name within the local sysplex
- domain ID within the local sysplex
- local sysplex name
- domain ID within the enterprise
- system name within the enterprise
- sysplex name within the enterprise

You have several choices as to how to route commands to other systems within the sysplex. These choices are described in the following subsections.

INGSEND

This will assist you in sending a command over a semi-shared RMTCMD session (this requires some configuration setup and depends upon VTAM).

1. Type **INGSEND** with no parameters on the command line of any NetView or system operations panel and you see the **INGSEND** panel:

```

INGKYSND                SA z/OS - Command Dialogs
Domain ID = IPUFM      ----- INGSND -----      Date = 06/09/04
Operator ID = OPER1                                     Time = 16:15:18

Specify or revise the following parameters:

Response      => YES           Enter ACK, YES, NO or CORR
Operator id   => OPER1         Leave blank for default
Target (TO)   => _____   format: [netid.]domain
Corrwait      => 10
Command ...

_____
_____
_____

Command ==>
PF1=Help    PF2=End    PF3=Return    PF6=Ro11    PF12=Retrieve
    
```

2. Fill out the other fields and press Enter. (For more information on the individual fields, refer to *System Automation for z/OS Operator's Commands*).

Communicating with Other Systems

Single System Image and Single Point of Control

If the command you want to send is an SA z/OS command and is supported by single system image (most of them are), you can just issue it, specifying the destination system as the target. This only requires that NetView be active on both systems - VTAM is not used.

The following example shows the usage of single system image to determine the status of APPL1 on every system within the sysplex (AOCA, AOCB, AOCC, AOCD) and then to restart it on AOCC where it is BROKEN. All operations are carried out from the IPUFA system.

```
DISPSTAT APPL1,TARGET=*ALL
```

```
AOFKSTA5          SA z/OS - Command Dialogs      Line 1    of 3
Domain ID   = IPUFM      ----- DISPSTAT -----   Date = 04/12/04
Operator ID = OPER1                                Time = 10:06:25
A ingauto B setstate C ingreq-stop D thresholds E explain F info G tree
H trigger I service J all children K children L all parents M parents
CMD RESOURCE      STATUS      SYSTEM      JOB NAME    A I S R D RS TYPE      Activity
-----
APPL1            AUTODOWN  AOCA        APPL1       Y Y Y Y Y Y TRANS  --none--
APPL1            DOWN      AOCB        APPL1       Y Y Y Y Y Y TRANS  --none--
APPL1            BROKEN    AOCC        APPL1       Y Y Y Y Y Y TRANS  --none--
APPL1            CTLDOWN  AOCD        APPL1       Y Y Y Y Y Y TRANS  --none--

Command ==>
PF1=Help   PF2=End   PF3=Return  PF4=INGLIST  PF5=Filters  PF6=Ro11
PF8=Forward PF9=Refresh PF10=Previous PF11=Next   PF12=Retrieve
```

Enter B against APPL1 on AOCC and press enter.

```
AOFKSA5          SA z/OS - Command Dialogs      Date = 04/12/04
Domain ID   = IPUFM      ----- SETSTATE -----   Time = 10:16:32
Operator ID = OPER1

Specify subsystem and new state:

Subsystem => APPL1      Subsystem to be processed
System    => AOCC      System name, domain ID, sysplex name or *ALL
State     =>           Put subsystem in the following state:
                        (RESTART / AUTODOWN / CTLDOWN / UP / ENDED /
                        MOVED)

Scope     => ONLY      Change state for:
                        ONLY - only this subsystem
                        CHILDREN - only dependant subsystems
                        ALL - resource and dependants

Start     => NO        Start or restart subsystem now (YES/NO)
Override  => NO        Override(Start=YES) (NO/FLG/TRG/SVP/EVT/ALL)

Command ==>
PF1=Help   PF2=End   PF3=Return  PF6=Ro11
PF8=Forward PF9=Refresh PF10=Previous PF11=Next   PF12=Retrieve
```

Complete the command parameters (STATE=RESTART, START=YES) and press ENTER to issue the command. It will be sent to the target system and executed.

Cross-Domain Sessions

Cross-domain sessions let you send commands to other domains using NetView RMTCMD, gateways, or Terminal Access Facility (TAF).

To display cross domain sessions, from the NetView command line, type A0C to invoke the SA z/OS main menu.

From the SA z/OS main menu, type 1 (Operator Interface) and press the Enter key. This takes you to the *Main Operator Menu*. On the command line, type 1 (CROSS DOMAIN) and press the Enter key. This leads to the *Cross Domain Menu*, as shown in Figure 99. From this menu you can display cross-domain sessions.

```
A0FK2D                SA z/OS - Operator Interface
Domain ID = IPUFM      ----- Command Domain Menu -----   Date = 06/09/04
Operator ID = OPER1                                         Time = 16:24:10

Select:

  1   Cross Domain - Enter Cross Domain Commands using INGSND
  2   Cross Domain - Enter Cross Domain Commands using TAF Sessions

Command ==>
PF1=Help      PF2=End      PF3=Return      PF6=Ro11
               PF12=Retrieve
```

Figure 99. Command Domain Menu Panel

Sending Commands to a Linux Target System

Use the ISQSEND command of processor operations to send a command to a Linux target system. The system previously must have been initialized successfully, using the ISQXIII command. In order to see the responses on the NetView screen, the command ISQXMON must have been used to add you to the interested operator list of the target system.

Since the Linux operating system handles commands case sensitive, you have to use the "netvasis" prefix. In the following example, the Linux command `ps -x` is issued to show Linux process related information. Note, that NetView suppresses the "netvasis" prefix when redisplaying the entered command.

```

NCCF                               Tivoli Netview  IPVFJ TIL      05/28/01 15:19:09
* IPVFJ  ISQSEND FCLX2 OC PS -X
U IPVFJ  ISQ901I FCLX2 SC ISQ417I CMD STATUS(ACCEPTED)
U IPVFJ  ISQ901I FCLX2 OC PS -X
U IPVFJ  ISQ017I ISQSEND COMPLETED SUCCESSFULLY.
U IPVFJ  ISQ901I FCLX2 OC PID TTY STAT TIME COMMAND
U IPVFJ  ISQ901I FCLX2 OC 1 ? S 0:00 init
U IPVFJ  ISQ901I FCLX2 OC 2 ? SW 0:00 ckMCHECK!
U IPVFJ  ISQ901I FCLX2 OC 3 ? SW 0:00 ckeventd!
U IPVFJ  ISQ901I FCLX2 OC 4 ? SW 0:00 ckswapd!
U IPVFJ  ISQ901I FCLX2 OC 5 ? SW 0:00 ckreclaimd!
U IPVFJ  ISQ901I FCLX2 OC 6 ? SW 0:00 cdbflush!
U IPVFJ  ISQ901I FCLX2 OC 7 ? SW 0:00 ckupdated!
U IPVFJ  ISQ901I FCLX2 OC 306 ? SW 0:00 ckeventd!
U IPVFJ  ISQ901I FCLX2 OC 309 ? SW 0:00 ckeventd!
U IPVFJ  ISQ901I FCLX2 OC 312 ? SW 0:00 ckeventd!
U IPVFJ  ISQ901I FCLX2 OC 315 ? SW 0:00 ckeventd!
U IPVFJ  ISQ901I FCLX2 OC 436 ? SN 0:00 syslogd -M 0
U IPVFJ  ISQ901I FCLX2 OC 450 ? SN 0:00 klogd
U IPVFJ  ISQ901I FCLX2 OC 469 ? SN 0:00 crond
U IPVFJ  ISQ901I FCLX2 OC 488 ? SN 0:00 inetd
U IPVFJ  ISQ901I FCLX2 OC 507 ? SN 0:00 httpD
U IPVFJ  ISQ901I FCLX2 OC 554 console S 0:00 -bash
U IPVFJ  ISQ901I FCLX2 OC 603 console R 0:00 ps -X
U IPVFJ  ISQ901I FCLX2 OC  root@boelnfc2 /root!#

-----
???
```

Figure 100. Sending a Command to a Linux Target System

Terminal Access Facility Full-Screen Sessions

The NetView terminal access facility (TAF) allows you to log on to multiple applications either on your system or other systems. *Tivoli NetView for z/OS Command Reference* describes in detail how to use TAF.

SA z/OS allows you to define TAF sessions in the customization dialog so that you do not have to set them up each time you want to use them. Refer to *System Automation for z/OS Defining Automation Policy* for more details.

Determining the Status of TAF Full-Screen Sessions

Using the TAF Fullscreen Menu panel you can see the list of TAF fullscreen sessions that have been defined to SA z/OS for your operator ID, and determine which of these are active.

To see the list of TAF fullscreen sessions available to you:

1. From the NetView command line, type **Oper**. You see the Main Operator Menu.
2. On the Selection line, type **1** (CROSS DOMAIN) and press the Enter key. You see the Cross Domain Menu.
3. On the Action line, type **2** and press the Enter key. You see the TAF Fullscreen Menu.

Note: If no TAF sessions are defined for your operator ID, you will see a line mode message

```
AOF041I UNABLE TO FIND FULL SESSION ENTRIES
```



```

A0FK2DF          SA z/OS - Command Dialogs          Line
Domain ID = IPUFM  --- TAF Fullscreen Display ---   Date = 05/11/04
Operator ID = OPER1      System = AOC7              Time = 16:34:53

Cmd:  I Initialize      R Return to      T Terminate

Cmd Name      Application  System    Status    SCRLU      Session id
-----
-   TSO        IPSFM     KEY3      ACTIVE    TAFFJF35
-   IMS        IPSFN     KEY4      INACTIVE
-----

Command ==>
PF1=Help      PF2=End          PF3=Return      PF6=Roll
                PF9=Refresh      PF12=Retrieve
    
```

Figure 101. TAF Fullscreen Display Panel

This panel shows the settings of all fullscreen sessions that are defined in the policy database.

Note: If you have defined exit routine AOFEXC04, there is an additional action code U. See *System Automation for z/OS Customizing and Programming* for details.

Managing TAF Full-Screen Sessions

You can use the *TAF Fullscreen Display* panel to log on to other applications on the current or another system to return to the session or to terminate the session.

For each TAF session the following information is displayed:

Name Is the name of the session.

Application

Is the name of the application as defined in the VTAMLST data set.

System

Is the name of the system where the application runs. This name is defined in the policy database.

Status Is the current status of the TAF session.

SCRLU

Is the logical unit name of the session.

Session ID

Is the session ID as assigned by NetView.

You can use the following command codes:

I Initializes the fullscreen session.

R Returns to session.

T Terminates the fullscreen session.

Gateway Sessions

In SA z/OS you can monitor other systems and issue commands to those systems using gateway sessions.

Gateways are useful because they:

- Save you the step of logging on to the NetView on the other system
- Let you communicate with another system even if you do not have an ID for that system
- Can provide a second NetView session with a system if you have an ID for that system

SA z/OS provides several panels that let you check your communication links with other systems and send commands to those systems. You can:

- Determine the status of gateway sessions
- Determine the status of paths. A path uses an intermediate system to connect two systems that are not directly linked
- Issue commands to other systems using gateway sessions

Determining the Status of Gateway Sessions

You can determine which gateway sessions have been defined and see which of these are active. You may want to do this if you use SDF to monitor other systems, since SDF uses gateway sessions to receive information from remote systems.

You also need to check the status of a gateway session if you want to receive notification messages from other systems. (Notification messages are messages that document significant actions that SA z/OS has detected or taken.) Finally, you need to check the status of a gateway session if you want to issue a command to another system using a gateway session.

You can obtain information on gateway sessions from the Gateway Display panel or from SDF, if SDF has been customized to do this.

To access the Gateway Display panel, type DISPGW from any command line and press the ENTER key.

```
AOFK2GL          SA z/OS - Command Dialogs          Line 1 of 6
Domain ID = IPUFM ----- DISPGW -----          Date = 11/04/03
Operator ID = LRM                                     Time = 09:38:06
          System = KEY3                               Focal Point = IPSFM
Domain  Sdfroot  Status  Description
-----
IPSFN   KEY4     ACTIVE  Gateway to target key4
IPSF0   KEY5     ACTIVE  Gateway to target key5
IPSF6   KEY6     ACTIVE  Gateway to target KEY6
IPUFG   AOC5     NOT STARTD Gateway to target AOC5
IPUFG   AOC7     ACTIVE  Gateway to target AOC7
IPVFG   TSCF2    ACTIVE  Gateway to target TSCF2
```

For further information on the DISPGW command refer to the *System Automation for z/OS Operator's Commands*.

Chapter 8. Scenario: Finding Out Why a Resource Is not Available

For most operators, the key resource to be monitored is the application. You want to be sure that your enterprise's critical jobs and subsystems are getting the service they require and are operating satisfactorily. The scenario contained in this chapter gives you a general idea of how to use SA z/OS commands to discover and react to problems affecting your important applications.

Imagine the following scenario:

You get a phone call from one of your users: Resource **MSCSK1** is not available. Please find out why and start it!

Here are the steps to make the resource available:

1. Use the INGLIST command to find out details (especially the status information) about resource *MSCSK1*. As you know the name of the resource, you can issue the command as follows on your NetView console:
INGLIST MSCSK1
2. Suppose you get an output screen as shown in Figure 102, which shows that the compound status for *MSCSK1* is **SATISFACTORY** which results from the fact that the observed status **SOFTDOWN** fulfills the desired status (**UNAVAILABLE**). However, you wonder why is the desired status **UNAVAILABLE** even though you and your user expected it to be available. This needs further investigation.

```
INGKYST0          SA z/OS - Command Dialogs          Line 1    of 1
Domain ID = IPUFM  ----- INGLIST -----          Date = 05/11/04
Operator ID = MSCH          Sysplex = AOC9PLEX          Time = 14:10:36
CMD: A Update   B Start    C Stop    D INGRELS  E INGVOTE  F INGINFO
      G Members  H DISPTRG  I INGSCHED J INGGROUP          / scroll
CMD Name      Type System  Compound  Desired   Observed  Nature
-----
E  MSCSK1     APL  AOC9    SATISFACTORY UNAVAILABLE SOFTDOWN

Which
votes are
involved?

Command ==>
PF1=Help   PF2=End   PF3=Return  PF4=DISPSTAT  PF5=Filters  PF6=Roll
PF9=Refresh PF10=Previous PF11=Next   PF12=Retrieve
```

Figure 102. INGLIST Output

3. To find out the votes that unexpectedly caused the resource to be unavailable (SOFTDOWN), you enter action code **E** into the **CMD** column next to the resource *MSCSK1* in the INGLIST command dialog panel as indicated in Figure 102 on page 191.
4. This will lead you to the INGVOTE command dialog as shown in Figure 103. From this panel you can see that there has been an operator request (MakeUnAvailable) with highest priority (hex 005A0000) to stop the resource; this has obviously been satisfied.

```

INGKYRQ0          SA z/OS - Command Dialogs          Line 1 of 4
Domain ID = IPUFM ----- INGVOTE -----         Date = 05/11/04
Operator ID = MSCH          Sysplex = AOC9PLEX          Time = 14:13:14

Resource ==> MSCSK1/APL/AOC9
System ==>          System name, domain id or sysplex name

Cmd: C cancel request  K Kill request  S show request details
Cmd Action WIN Request/Vote Data
-----
C   STOP   Y Request   : MakeUnAvailable
      Originator : OPERATOR(MSCH)
      Priority   : 005A0000   Must Be Down - Operator
      Status    : Winning/Satisfied

Command ==>
PF1=Help      PF2=End          PF3=Return      PF6=Ro11
              PF9=Refresh     PF12=Retrieve

```

Figure 103. INGVOTE Output

5. As this request is persistent and can only be beaten by any other request with priority **FORCE**, you must cancel it to allow the resource to become available again. (The request obviously stems from an operator's INGREQ command with high priority which is the strongest request you can issue with SA z/OS, see also "Automation Concepts" on page 13 for the documentation of the hierarchy of requests). Therefore you enter action code **C** into the **CMD** column next to the request as indicated in Figure 103.
6. On the INGVOTE command dialog, you will see the confirmation that the request has been cancelled. Now press PF3 to go back to the INGLIST command dialog and press the PF9 key to refresh the information shown for resource *MSCSK1*. Let us assume you see the output as shown in Figure 104 on page 193.

```

INGKYST0          SA z/OS - Command Dialogs      Line 1 of 1
Domain ID = IPUFM ----- INGLIST -----      Date = 05/11/04
Operator ID = MSCH          Sysplex = AOC9PLEX      Time = 14:16:00
CMD: A Update   B Start   C Stop   D INGRELS   E INGVOTE   F INGINFO
      G Members   H DISPTRG  I INGSCHED J INGGROUP      / scroll
CMD Name      Type System   Compound   Desired   Observed   Nature
-----
F  MSCSK1     APL AOC9     AWAITING  AVAILABLE  SOFTDOWN

                                Refresh INGLIST
                                status with PF9

Command ==>
PF1=Help   PF2=End   PF3=Return  PF4=DISPSTAT  PF5=Filters  PF6=Ro11
PF9=Refresh PF10=Previous PF11=Next   PF12=Retrieve

```

Figure 104. Refreshed INGVOTE Output

What you now see is that the compound status for *MSCSK1* is **AWAITING** (awaiting automation) and the observed status is still **SOFTDOWN** which is not exactly what you might have wanted or expected. You want observed and desired status to be **AVAILABLE**. There must be something else that inhibits this. Maybe there are dependency relationships to supporting resources that are not fulfilled.

7. You now use INGINFO (action code **F**) to display more information about why *MSCSK1* is still awaiting automation. Scroll down the INGINFO output panel until you reach a panel similar to the one shown in Figure 105.

```

INGKYIN0          SA z/OS - Command Dialogs      Line 29 of 145
Domain ID = IPUFM ----- INGINFO -----      Date = 05/11/04
Operator ID = MSCH          Sysplex = AOC9PLEX      Time = 14:17:37

Resource ==> MSCSK1/APL/AOC9          format: name/type/system
System ==>          System name, domain ID or sysplex name

Schedule          : -None-

Backward Relationships :
  AOC9/SYG/AOC9          HasMember

Forward Relationships :
  MSCSK2/APL/AOC9          HasParent
  NETVIEW/APL/AOC9          HasParent

Requests          :

Votes             :

Command ==>
PF1=Help   PF2=End   PF3=Return  PF4=DISPINFO  PF6=Ro11
PF7=Back   PF8=Forward PF9=Refresh  PF12=Retrieve

```

Figure 105. INGINFO Output

From this output panel you find out that *MSCSK1* has a *HasParent* dependency on the supporting resource *MSCSK2/APL/AOC9*.

Figure 106 on page 194 illustrates the current situation as you have investigated it until now using the various commands: An operator has made

MSCSK1 unavailable by issuing a *SOFTDOWN* request. *MSCSK1* is dependent upon *MSCSK2*.



Figure 106. Current Automation Situation

- You now need to find out more about *MSCSK2*. To do this, proceed as in step 1 on page 191 using the *INGLIST* command:

```
INGLIST MSCSK*
```

You may get an output similar to the one shown in Figure 107

```

INGKYST0          SA z/OS - Command Dialogs          Line 1 of 2
Domain ID = IPUFM ----- INGLIST ----- Date = 05/11/04
Operator ID = MSCH Sysplex = AOC9PLEX Time = 14:19:32
CMD: A Update B Start C Stop D INGREL E INGVOTE F INGINFO
      G Members H DISPTRG I INGSCHED J INGGROUP / scroll
CMD Name Type System Compound Desired Observed Nature
-----
MSCSK1 APL AOC9 AWAITING AVAILABLE SOFTDOWN
E MSCSK2 APL AOC9 SATISFACTORY UNAVAILABLE SOFTDOWN

Which
votes are
involved?

Command ==>
PF1=Help PF2=End PF3=Return PF4=DISPSTAT PF5=Filters PF6=Roll
PF9=Refresh PF10=Previous PF11=Next PF12=Retrieve

```

Figure 107. *INGLIST* Output for *MSCSK** Resources

This panel informs you that *MSCSK2* is also not available (observed status is *SOFTDOWN*), and that this even conforms to its desired status. This is obviously not what you want, so now you must find out the reason for this.

- As in step 3 on page 192, you use action code *E* to find out the votes that unexpectedly caused the *MSCSK2* resource to be unavailable (*SOFTDOWN*). *SA z/OS* might display a panel like shown in Figure 108 on page 195.

```

INGKYRQ0          SA z/OS - Command Dialogs          Line 1 of 4
Domain ID = IPUFM ----- INGVOTE -----          Date = 05/11/04
Operator ID = MSCH          Sysplex = AOC9PLEX          Time = 14:20:43

Resource ==> MSCSK2/APL/AOC9
System ==>          System name, domain id or sysplex name

Cmd: C cancel request K Kill request S show request details
Cmd Action WIN Request/Vote Data
-----
STOP Y Request : MakeUnAvailable_Only
      Originator : SCHEDULE
      Priority : 00520000 Must Be Down - Schedule
      Status : Winning/Satisfied

Command ==>
PF1=Help PF2=End PF3=Return PF6=Roll
PF9=Refresh PF12=Retrieve

```

Figure 108. INGVOTE Output for MSCSK2

From this panel you can see that there has been a request (MakeUnAvailable) generated by a service period (SCHEDULE) to stop the resource MSCSK2. This obviously has been satisfied.

- To find out more about the service period defined for MSCSK2, on the INGLIST panel, type action code I into the CMD column next to MSCSK2. Suppose, you see a display as shown in Figure 109. From this figure, you can see that the resource MSCSK2/APL/AOC9 is scheduled with high priority to be down on Tuesdays between 14:10 PM and 16:00 PM and that the current time is 14:21:50, which is actually within the requested downtime (and indeed, the indicated date of 05/11/04 was a Tuesday).

```

INGKYSP2          SA z/OS - Command Dialogs          Date = 05/11/04
Domain ID = IPUFM ----- INGSCHED -----          Time = 14:21:50
Operator ID = MSCH          Sysplex = AOC9PLEX

Resource . . . : MSCSK2/APL/AOC9          Schedule . . . : MSCSCED8
Starting date => 05 / 09 / 2004          (mm/dd/yyyy)

C Date          Pri Strt/Stop Strt/Stop Strt/Stop Strt/Stop Strt/Stop
-----
05/11/04 Tue UP H 1320 1340 1400 1402
          DN H 1300 1319 1341 1359 1410 1600
05/12/04 Wed UP H 1320 1340 1400 1402
          DN H 1300 1319 1341 1359
05/13/04 Thu UP H 1320 1340 1400 1402
          DN H 1300 1319 1341 1359
05/14/04 Fri UP H 1320 1340 1400 1402
          DN H 1300 1319 1341 1359
05/15/04 Sat UP H 1320 1340 1400 1420 1500 1520 1540 1600 1640 2359
          DN H 1300 1319 1341 1359 1440 1459 1521 1539 1620 1639

Command ==>
PF1=Help PF2=End PF3=Return PF6=Roll
PF7=Backward PF8=Forward PF9=Refresh PF12=Retrieve

```

Figure 109. INGSCHED for MSCSK2

- To allow automation to make the MSCSK2 available and keep it available for the rest of the day, you could now specify an additional uptime schedule for

Tuesdays, starting from now (14:26 PM) until midnight (24:00) as shown in Figure 110. This method is called a schedule override (as it overrides a schedule defined by automation policy).

Note:
Schedule overrides done by INGSCHED are permanent.

```

INGKYSP2          SA z/OS - Command Dialogs
Domain ID = IPUFM ----- INGSCHED ----- Date = 05/11/04
Operator ID = MSCH Sysplex = AOC9PLEX Time = 14:25:18

Resource . . . : MSCSK2/APL/AOC9 Schedule . . . : MSCSCED8
Starting date => 05 / 09 / 2004 (mm/dd/yyyy)

C Date          Pri Strt/Stop Strt/Stop Strt/Stop Strt/Stop Strt/Stop
-----
05/11/04 Tue UP H 1320 1340 1400 1402 1426 2400
                   DN H 1300 1319 1341 1359 1410 1426
05/12/04 Wed UP H 1320 1340 1400 1402
                   DN H 1300 1319 1341 1359
05/13/04 Thu UP H 1320 1340 1400 1402
                   DN H 1300 1319 1341 1359
05/14/04 Fri UP H 1320 1340 1400 1402
                   DN H 1300 1319 1341 1359
05/15/04 Sat UP H 1320 1340 1400 1420 1500 1520 1540 1600 1640 2359
                   DN H 1300 1319 1341 1359 1440 1459 1521 1539 1620 1639

AOF099I FUNCTION SUCCESSFULLY COMPLETED
Command ==>
PF1=Help PF2=End PF3=Return PF6=Ro11
PF7=Backward PF8=Forward PF9=Refresh PF12=Retrieve

```

Figure 110. Schedule Override for MSCSK2

12. Now with PF3, go back to the INGLIST command dialog and press PF9 to see if any changes have occurred. From Figure 111 you can see that MSCSK2 is now available, but MSCSK1 is still SOFTDOWN, AWAITING automation.

```

INGKYST0          SA z/OS - Command Dialogs Line 1 of 2
Domain ID = IPUFM ----- INGLIST ----- Date = 05/11/04
Operator ID = MSCH Sysplex = AOC9PLEX Time = 14:26:23
CMD: A Update B Start C Stop D INGRELS E INGVOTE F INGINFO
      G Members H DISPTRG I INGSCHED J INGGROUP / scroll
CMD Name Type System Compound Desired Observed Nature
-----
MSCSK1 APL AOC9 AWAITING AVAILABLE SOFTDOWN
MSCSK2 APL AOC9 SATISFACTORY AVAILABLE AVAILABLE

Command ==>
PF1=Help PF2=End PF3=Return PF4=DISPSTAT PF5=Filters PF6=Ro11
PF9=Refresh PF10=Previous PF11=Next PF12=Retrieve

```

Figure 111. Refreshed INGLIST Output

As a consequence, you might suppose that for this resource, some automation flag inhibits startup or automation.

13. So in a next step, you check the automation flags for MSCSK1. You can do this by scrolling the INGLIST output from Figure 111 to the right with PF11 until you reach a display similar to Figure 112 on page 197.


```

INGKYST0          SA z/OS - Command Dialogs      Line 1    of 2
Domain ID = IPUFM ----- INGLIST -----      Date = 05/11/04
Operator ID = MSCH          Sysplex = AOC9PLEX      Time = 14:27:20
CMD: A Update  B Start  C Stop  D INGRELS  E INGVOTE  F INGINFO
      G Members  H DISPTRG  I INGSCHED  J INGGROUP  / scroll
CMD Name      Type System  Automation  Startable  Auto Hold  Category
-----
MSCSK1       APL  AOC9    IDLE       YES        YES NO
MSCSK2       APL  AOC9    IDLE       YES        YES NO

Command ==>
PF1=Help    PF2=End      PF3=Return  PF4=DISPSTAT  PF5=Filters  PF6=Ro11
PF9=Refresh PF10=Previous PF11=Next   PF12=Retrieve

```

Figure 112. Checking Automation Flags with INGLIST

However, the result shows that *MSCSK1* is startable and automation is allowed for it (*Auto=YES* and *Hold=NO*). So what else can be the reason that prevents *MSCSK1* from starting?

14. You probably guessed right: there might be a trigger for *MSCSK1* with the events for its startup condition not set, thus preventing the resource from starting. Scroll horizontally in your current INGLIST output until you see the column **Trigger** as shown in Figure 113.

Note: The DISPSTAT command cannot be used as it only displays service periods and triggers from downlevel SA z/OS systems.

```

INGKYST0          SA z/OS - Command Dialogs      Line 1    of 2
Domain ID = IPUFM ----- INGLIST -----      Date = 05/11/04
Operator ID = MSCH          Sysplex = AOC9PLEX      Time = 14:28:33
CMD: A Update  B Start  C Stop  D INGRELS  E INGVOTE  F INGINFO
      G Members  H DISPTRG  I INGSCHED  J INGGROUP  / scroll
CMD Name      Type System  Starttype  Stoptype  Trigger  Schedule
-----
MSCSK1       APL  AOC9                MSCTRG1
MSCSK2       APL  AOC9                MSCSCED8

Command ==>
PF1=Help    PF2=End      PF3=Return  PF4=DISPSTAT  PF5=Filters  PF6=Ro11
PF9=Refresh PF10=Previous PF11=Next   PF12=Retrieve

```

Figure 113. Checking Triggers with INGLIST

From this panel, you see that indeed, a trigger *MSCTRG1* is connected to *MSCSK1*.

Now let us have a look at Figure 114 on page 198 to see the automation scenario for resource *MSCSK1*:

- *MSCSK1* is dependent upon *MSCSK2*. So *MSCSK2* must be up before *MSCSK1* can be started.
- Availability of *MSCSK2* is controlled by a schedule. So the time of day must be within one of the uptime schedules defined for this resource.
- *MSCSK1* has a connected trigger. On top of all the other prerequisites for starting this resource, at least one startup condition for that trigger must be fulfilled to make this resource available.

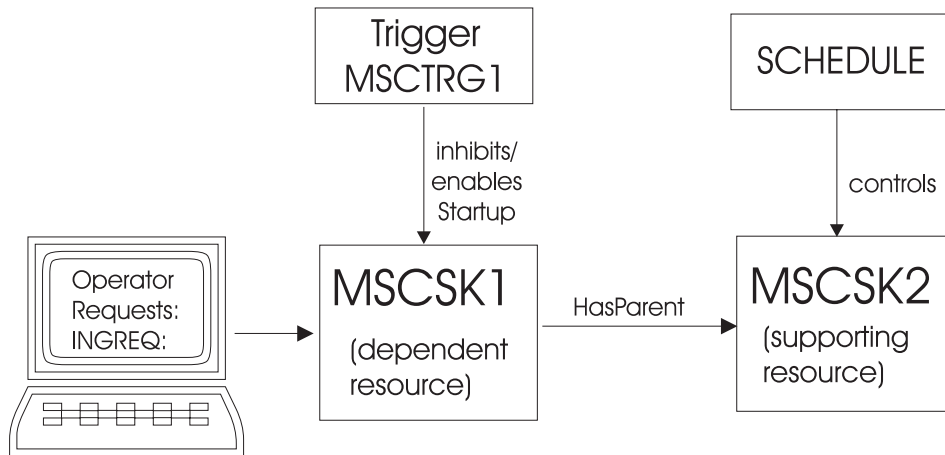


Figure 114. Complete Automation Environment for MSCSK1

- Now in this step, you will check the startup conditions for trigger *MSCTRG1*. You achieve this by entering action code **H** (for *DISPTRG*) into the **CMD** column next to *MSCSK1*. This will display triggers and their events for the selected resource. In this scenario, we get the *DISPTRG* output shown in Figure 115.

```

AOFLT000          SA z/OS - Command Dialogs          Line 1    of 2
Domain ID   = IPUFM          ----- DISPTRG -----   Date = 05/11/04
Operator ID = MSCH          System = AOC9              Time = 14:29:47

Resource ==> MSCSK1/APL/AOC9
System ==>          System name, domain id or sysplex name
Trigger. . . : MSCTRG1          Observed status : SOFTDOWN

Cmd: S show details
Cmd Type      Events
-----
S  STARTUP    MSCEVTUP
   SHUTDOWN   MSCEVTDN

Command ==>
PF1=Help      PF2=End          PF3=Return      PF6=Roll
               PF9=Refresh      PF12=Retrieve
  
```

Figure 115. *DISPTRG* Command Dialog

In this output, you can see that trigger *MSCTRG1* has a startup condition, which will be fulfilled if event *MSCEVTUP* is set. This is what you need to find out in the next step.

- To see more details about an event (or about the condition), type action code **S** into the **CMD** column next to the startup condition (event *MSCEVTUP*). This produces the following output:

```

AOFLT100          SA z/OS - Command Dialogs          Line 1 of 1
Domain ID = IPUFM  --- Trigger Condition List ----   Date = 05/11/04
Operator ID = MSCH          System = AOC9             Time = 14:30:46

Trigger . . . . : MSCTRG1
Condition type. : STARTUP
Resource. . . . : MSCSK1/APL/AOC9                    Observed status : SOFTDOWN

Cmd: D show resources  S set event for resource  U unset event for resource
Cmd Event  Status      Unset  Description
-----
S  MSCEVTUP  UNSET    UP    scheduling test event UP

Command ==>
PF1=Help    PF2=End          PF3=Return      PF6=Roll
              PF9=Refresh      PF12=Retrieve

```

Figure 116. Trigger Condition List

17. From Figure 116, we see at a glance that the event *MSCEVTUP* is not set (column **Status** contains UNSET). By typing action code **S** (set event for resource) into the **CMD** column next to the event *MSCEVTUP*, you now set this event to fulfill the startup condition for trigger *MSCTRG1*, thus allowing resource *MSCSK1* to start.

By magic, SA z/OS will now put your environment into the status you desired.

After a few minutes, you should be able to verify this. Go back to the INGLIST command dialog for both resources involved (INGLIST MSCSK*):

```

INGKYST0          SA z/OS - Command Dialogs          Line 1 of 2
Domain ID = IPUFM  ----- INGLIST -----         Date = 05/11/04
Operator ID = MSCH          Sysplex = AOC9PLEX      Time = 14:32:52
CMD: A Update      B Start      C Stop      D INGRELS   E INGVOTE   F INGINFO
      G Members    H DISPTRG  I INGSCHED  J INGGROUP  / scroll
CMD Name          Type System  Compound    Desired     Observed    Nature
-----
MSCSK1           APL  AOC9    SATISFACTORY AVAILABLE AVAILABLE
MSCSK2           APL  AOC9    SATISFACTORY AVAILABLE  AVAILABLE

```

Command ==>
PF1=Help PF2=End PF3=Return PF4=DISPSTAT PF5=Filters PF6=Roll
PF9=Refresh PF10=Previous PF11=Next PF12=Retrieve

Figure 117. Verifying Automation Status

Now you see that both resources have AVAILABLE for the observed and desired status which is SATISFACTORY for SA z/OS as the compound status and, hopefully, for you , too.

Chapter 9. Using I/O Operations

I/O operations is a highly flexible connection environment that expands the distance for channel and control unit communication.

You can use the SA z/OS I/O operations component:

- to get a single point of logical control for managing connectivity in your active I/O configurations.
- to have information presented on the operator console from which you issue a command or from the I/O Operations ISPF dialog. You can also use the I/O operations application program interface (API) to get information via a user-written call.
- as a tool for automating these I/O configurations. I/O operations takes an active role in detecting unusual conditions and lets you view and change paths between a processor and an I/O device, using dynamic switching (the ESCON director).

To fulfil the tasks of managing and automating I/O configurations, various commands are available, like for example, a command to define a dynamic connection between ports, commands to block or permit data transfer through ports or a series of commands to display I/O configuration information.

You can issue these commands either through the ISPF dialog (see Appendix B, “Using the I/O Operations ISPF Dialog,” on page 263) or from the operator console.

More features of the I/O operations are, for example:

- For an I/O resource that passes through a dynamic switch, I/O operations can detect and report several types of I/O configuration mismatches. It reports these mismatches when the program is started or when the configuration is dynamically reconfigured.
- You can use I/O operations as a single point of control:

I/O operations lets you change the connectivity of I/O resources. From any one system where I/O operations is running, you may affect the I/O connectivity of a target system on which I/O operations is also operating, thereby providing a single point of control.

I/O operations performs these functions when you enter, or send, one of its connectivity commands to one of the hosts on which the program is running. That host issues the commands to the affected hosts. The hosts that can be affected by the command depend on several factors, including the domain of the issuing host.

With the feature of a single point of control, you can:

- remove the dependency on a dynamic switch
- manage the devices across hosts; that is, you can vary devices on or off
- remove and restore channel paths among system images and have the pathing status automatically adjusted
- manage switchable paths in a system-integrated manner

With I/O operations, you can display and control the following I/O resources:

- *Channels* of the following types:

Making Connectivity Changes

- ESCON channels
- Parallel channels
- Coupling facility channels
- *Dynamic switches*, such as the ESCON Directors
- *Ports*, which are the interfaces on a dynamic switch
- *Control units*
- *I/O devices*
- and other objects

Making Connectivity Changes

With I/O operations you can easily perform the following connectivity changes:

- Remove and restore channel paths: These commands not only instruct z/OS to reconfigure a CHPID, they also issue the appropriate vary path requests. In these commands, you can specify any CHPID that is defined to one of the hosts in the issuing host's group. The CHPID can be either switchable or point-to-point, ESCON or parallel.
- Remove and restore devices: These commands allow you to remove one, a range, or an array of devices from - and restore them to - one or an array of hosts in a group. These commands therefore expand the capabilities of the analogous Vary Device system commands, so that devices can be varied across system images—all within the issuing host's group.
- Use switching commands, like:
 - Commands that affect only ports
 - Remove Switch
 - Restore Switch
 - Commands that define a chained path and remove these definitions

Display Commands To Obtain Data About I/O Resources

With such commands, you can for example, retrieve data about one or all channels (CHPIDs) defined to the host image on which the command was issued. Or you can retrieve data about one, a range, or all the devices and their paths defined to the host image. You can also display data about the addressable port(s) on one or all switches allocated, or attached, to the issuing host.

Examples for this category of commands are:

- DISPLAY CHP
- DISPLAY DEVICE
- DISPLAY PORT
- DISPLAY RESULTS

Connectivity Commands That Affect a Switch

With commands of this category you can for example permit or prohibit dynamic communication between two specified ports or between one port and all other ports on the same switch or the same ports on all the switches allocated to, or attached to, the issuing host.

Connectivity Commands That Affect a Switch

Or, you can establish or remove a static, or dedicated, connection between two ports on the same switch or the same two ports on all the switches allocated to, or attached to, the issuing host.

Examples for this category of commands are:

- ALLOW
- PROHIBIT
- CONNECT
- DISCONNECT
- BLOCK
- UNBLOCK
- CHAIN
- UNCHAIN

Display and Utility Commands Related to Connectivity Commands

With commands of this category you can for example display the specified number or all of failed vary path requests as a result of the last connectivity command.

Or you can build a list of the vary path requests that would be needed to synchronize path status with the physical settings of the specified switch.

You can also display or reset the number of seconds that the issuing hosts (I/O operations) will wait to receive votes from the I/O operations that from its voting set.

Examples for this category of commands are:

- DISPLAY VARY
- SYNC SWITCH
- RESET HOST
- DISPLAY TIMEOUT
- RESET TIMEOUT

Utility Commands for General Processing

With such commands, you can for example, acquire or release exclusive control of I/O operations command processing for a specified user ID.

Or you can instruct the switch(es) to send their log data to the host on which the issuing I/O operations is running.

Another usage of such a command would be to ensure that the issuing I/O operations's configuration map is correlated with the active configuration on the switch(es).

Examples for this category of commands are:

- GETLOCK
- UNLOCK
- LOGREC
- RESET SWITCH

Using the I/O Operations ISPF Dialog

A set of REXX EXECs is shipped with SA z/OS that – when installed as the I/O operations Interactive System Productivity Facility (ISPF) dialog – offers you a simple way to enter the same commands to I/O operations that you can enter at an z/OS operator console.

In addition to entering I/O operations commands, you can create entire switch configurations, starting with the default matrix of switch ports that is shipped with the dialog.

Detailed information about the I/O operations ISPF dialog is included in Appendix B, “Using the I/O Operations ISPF Dialog,” on page 263. The information in this section helps you get started using the dialog. For more information, refer to the online tutorial and online help panels supplied with the dialog.

Using the z/OS Operator Console for I/O Operations Commands

Some I/O operations commands can be entered at the z/OS operator console. To enter a command in the z/OS environment, use the *Modify* command, which can be shortened to F. Specify the I/O operations procedure name that has been created as a member in your SYS1.PROCLIB. The member name for the procedure as supplied by SA z/OS is *INGEIO*.

- If you did not append an z/OS identifier to the Start command when I/O operations was started, the syntax is as follows:

```
/modiFy procname,I/O_operations_command
```

- If you did append an z/OS identifier to the Start command when I/O operations was started, the syntax is as follows. (NOTE: if the SA z/OS I/O operations component was started automatically by the automation supplied with SA z/OS, an identifier of "IO" was used by the Start command.)

```
/modiFy identifier,I/O_operations_command
```

For example, */F IO,D N CU.C00* ** instructs I/O operations to display names of, and salient data about, the port or ports that have been assigned names with the common character string 'CU.C00' on the dynamic switches allocated to the issuing instance of I/O operations.

The NetView program provides an z/OS interface that allows a NetView REXX EXEC or an operator at a NetView console to issue z/OS operator commands. For example, the following Display Switch command can be coded in a REXX EXEC or issued at a NetView console:

```
MVS /F IO,D S *
```

(The output of this Display Switch command is displayed on the NetView console, not the z/OS operator console.)

When you use the NetView console to enter I/O operations commands, keep the following points in mind:

- Control authorization is not checked. The same I/O operations commands can be entered that can be entered at the z/OS operator console.

Using the z/OS Operator Console for I/O Operations Commands

- Commands can be entered from different consoles to the same system operator ID. You should, therefore, have a central point of control for I/O operations processing, especially for its connectivity and utility commands.

Using the z/OS Operator Console for I/O Operations Commands

Chapter 10. VM Second Level Systems Support

This feature provides ProcOps support to control and monitor guest machines running under VM.

ProcOps allows an operating system to be IPLed into a processor, amongst other facilities. Such an operating system is VM. Within VM other operating systems can be IPLed as guest machines. Of particular interest are LINUX guest machines, but MVS, VSE and even VM guest machines may be possible. (Lower levels of guest machines are not considered). Previously there was no effective way to enter commands to and receive messages from such a guest target system in order to validate that it had IPLed correctly, or that it is behaving correctly.

With second level guest machine support you can:

- Capture messages issued by the guest machine itself and route these back to the ProcOps process for display or automated processing, or both
- Send commands to the guest machine from ProcOps, either as operator requests or automated actions

ProcOps Dialog

The ProcOps dialogs flow has been enhanced to provide you with information about the guest systems hierarchy (see Figure 118):

- Hosting systems are flagged on the Status Summary panel ISQESUM.
- The PSMs of a hosting target system can be displayed by selecting the Target System Summary panel ISQETARG and the following PSMs panel ISQETRGG. The Target Hardware Summary panel for the PSM shows the initialized guest systems per PSM.
- The hosting target system of a guest system can be displayed by selecting the Target Hardware Summary panel ISQEHARP.

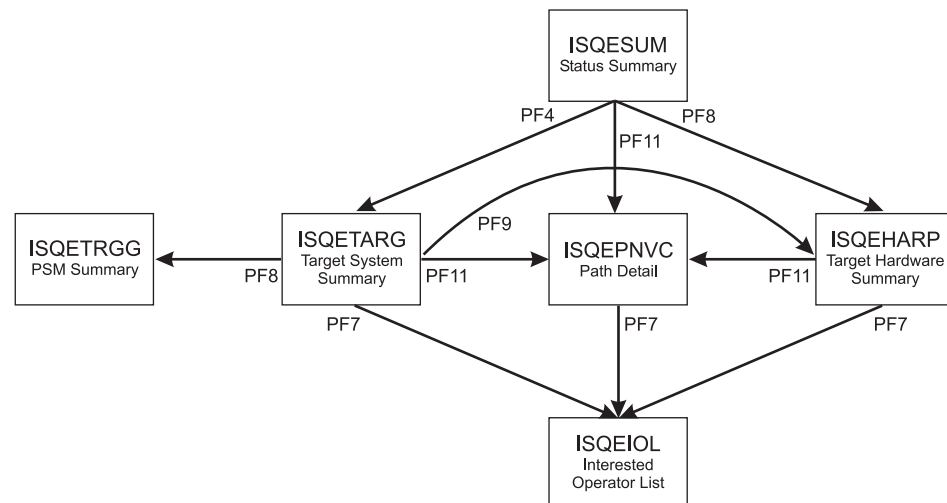


Figure 118. ProcOps Panel Flow

Status Summary Panel (ISQESUM)

```

ISQESUM      SA z/OS Proc-Ops Target Status Summary      Updates: Dynamic
Control file in use: SATEST.OA02311.ISQCNTL
NMC Bridge: INACTIVE
I isqxiii C isqxcls 0 isqxopt  A,B view netlogs E events  Debug Mode: OFF

Cmd   Target System  Status      Focal Points - Primary: IPSFO Backup: IPUFM
BOEVMRAP (*)  UNKNOWN
CFA      CLOSED
CFB      INITIALIZED
CFF      INITIALIZED
CF1      INITIALIZED
CF61     INITIALIZED
CF62     INITIALIZED
KEYA     INITIALIZED
KEYB     INITIALIZED
KEYC     NOT ACTIVE
KEYZ     NOT OPERATING
KEY1     INITIALIZED
KEY2     INITIALIZED
KEY3     CLOSED
KEY4     INITIALIZED
KEY6     INITIALIZED
KEY6B    UNKNOWN
KEY7     NOT OPERATING
LNXOLI1  CLOSED
TEL01FE  CLOSED
TEL17FE  UNKNOWN
TEL30FE  UNKNOWN

Enter=Static PF1=Help PF3=Exit PF4=Tgt Sys Summary PF5=Debug On/Off
PF6=Roll    PF7=Up   PF8=Down PF9=Tgt HW Summary PF11=PATH Details PF12=Quit
  
```

Figure 119. Status Summary Panel ISQESUM

Targets that host other target systems are marked with (*) that is *hosting system*.

Target Hardware Summary Panel (ISQEHARP)

```

ISQEHARP      SA z/OS Proc-Ops THW Summary      Updates: Dynamic

Target Hardware Name: YORAMA      Type: PSM
VM Host System Name: VM123
Target Hardware Description: ProcOps Service Machine for KEYn systems
Initialized Target Systems:
KEY6      KEY7      KEY8

Communication Path Error Status:

Last Significant Message:

Enter=Static PF1=Help PF3=Exit PF6=Roll PF7=Oper1st PF11=PATH DtlS PF12=Quit
  
```

Figure 120. Target Hardware Summary Panel for PSMs ISQEHARP

```
ISQEHARP          SA z/OS Proc-Ops THW Summary          Updates: Dynamic
Target Hardware Name: YORAMA      Type: eServer      Mode: LPAR
                                Model: RX4              Serial Number: 51-0064523
Target Hardware Description: Host for VM systems
Initialized Target Systems:
SYS1      (*) SYS2          SYS3

Communication Path Error Status:

Last Significant Message:

Enter=Static PF1=Help PF3=Exit PF6=Roll PF7=Oper1st PF11=PATH DtlS PF12=Quit
```

Figure 121. Target Hardware Summary Panel for Non-PSMs ISQEHARP

Processor Type PSM indicates that it is a ProcOps Service Machine and that the initialized target systems are Guest systems.

For PSMs:

- The Hosting System Name is displayed
- Mode, Model, and Serial Number are not displayed.

For real hardware:

- Initialized Target Systems that are hosting target systems are flagged with an (*).

Target System Summary Panel (ISQETARG)

```
ISQETARG          SA z/OS Proc-Ops Target System Summary      Updates: Dynamic

Target System
  Name           : KEY6____ (*)
  Description    :
  O.S.          : MVS
  Load Profile   :
  Status        : INITIALIZED
  Attention     : EXCEPTIONS

Target Hardware
  Name           : YORAMA           Mode : LPAR
  Attention     :
  Path Status   : ACTIVE

Target Hardware LPAR
  Name           : KEY6           Mode : ESA
  Image Profile : KEY6

Last Significant Message:

Enter=Static PF1=Help PF3=Exit PF6=Roll PF7=Oper List PF8=Guests
PF9=Target Hardware PF11=Path Detail PF12=Quit
```

Figure 122. Target System Summary Panel ISQETARG

If the target system is a hosting system then:

- (*) = Hosting system is displayed
- PF8=Guests is displayed. PF8 displays panel ISQETRGG.

Guests Summary Panel (ISQETRGG)

```
ISQETRGG          SA z/OS Proc-Ops PSMs/Guests Summary      Updates: Dynamic

VM Host System Name:

Defined ProcOps Service Machines:
  PSM01         PSM02

Initialized Target Systems:
  PSM01/Linux1  PSM01/Linux2  PSM01/Linux3  PSM02/MVS1
  PSM02/VSE1

Enter=Static PF1=Help PF3=Exit PF6=Roll PF7=Up PF8=Down PF12=Quit
```

Figure 123. PSMs/Guests Summary Panel ISQETRGG

This panel displays the guest (that is, hosted) systems for a target system which is a hosting system. The guest systems are displayed for each associated PSM.

VM Host System Name

Displays the name of the hosting system.

Defined ProcOps Service Machines

Lists PSMs defined in the control file to run on the hosting system.

The number of PSMs per hosting VM system is limited to 12. (This is checked by the configuration dialog.)

Initialized Target Systems

Lists target systems currently associated with the hosting system. The target systems are identified by PSM/Target.

The number of guest systems being controlled by a PSM is limited to 60. (This is checked by the configuration dialog.) Therefore the total number of guest systems running on a VM host is limited to 720.

PATH Detail Panel (ISQEPNVC)

```
ISQEPNVC          SA z/OS Proc-Ops  PATH Detail          Updates: Dynamic
PSM Name          : SAFOS              Connection Type      : TCPIP
Message Server Port : 4711_           Command Server Port : 4712_
IP Stack          : STACK---
PSM IP Address or Hostname:
999.999.999.999

Security Resource Name : DEIBMPI1.IP3T1200
Connection Authorization: PROCOPS      Lock Holder       :

PATH Status      :

Poll Frequency   : 40
Poll Retries     : 2

Last Significant Message:

Enter=Static      PF1=Help         PF3=Exit          PF6=Roll
PF7=Oper List    PF12=Quit
```

Figure 124. PATH Detail Panel ISQEPNVC

For PSM hardware TCPIP address, Message Server port, Command Server port, IP stack, and Security resource name are displayed.

Security Resource Name

Identifier used for SAF checking (for example, by RACF).

PSM Console Mode

Normally the PSM runs disconnected. In this mode, its regular, unsolicited messages are not seen, even though they are recorded by the Logger thread in its log data set. To assist problem determination, these regular messages are captured by the Message Handler, queued to the Message Queue, and routed to the current ProcOps focal point by the Message Server. They should then be seen in the NetView log.

When a user logged on to the PSM issues a PSM request, a CP command or a CMS command, the responses are always displayed to the user.

So, if a user logs on to the PSM no unsolicited messages are seen. These unsolicited messages can be made visible by issuing the command:

```
CP SET VMCONIO OFF
```

on the VM console. If the Message Handler is restarted for any reason (for example, when the first guest machine on the host is initialized using the ISQXIII command), the unsolicited messages to the VM console may be again lost. Reissue the CP SET command to restore message display at the VM console.

If an ISQTRACE command is issued that sets any thread trace to the ON state, the messages issued by the PSM are automatically displayed at the VM console, and not sent to NetView. This is to avoid possible message loops, because each message sent may cause additional messages to be created.

Note that if indentation of messages is active, the PSM messages returned to NetView also appear indented. If any PSM message spills to another line, that new line is a separate message sent to NetView.

If an ISQTRACE command is issued, and all traces are set to the OFF state, the flow of unsolicited messages is once again routed to NetView. This means that the effect of any earlier CP SET VMCONIO OFF command is then lost.

When messages are displayed on the VM console, they may also be spooled to a VM output file or a reader file. This includes any trace messages.

When a user is logged on to the PSM virtual machine, the following points should be noted.

- If messages are sent to the VM console they may cause the PSM to halt execution temporarily when or shortly after the screen fills with messages. Consider entering the CP TERMINAL MORE command to control the effect of this (possibly as a command in an initial EXEC that invokes the PSM Control Program). For example, issuing CP TERMINAL MORE 0 0 means that the VM console never locks up when regular messages are issued.
- Some messages are highlighted and held. For example, the XAUTOLOG command causes such messages. When the screen fills (even if CP TERMINAL MORE has been issued) the session goes into HOLDING mode and requires operator intervention.
- Any CP command can be executed by prefixing it with CP. Any CMS command can be executed by prefixing it with CMS. However any command is executed by the Command Server thread and serializes that thread. If such a command requires an extended period of time to execute it can prevent the Command Server thread from responding to requests. If a full screen command (such as XEDIT) is entered, all threads of the PSM Control Program stop executing until the user leaves the full screen command.

Starting the PSM Control Program

The PSM Control Program runs in its own virtual machine. To start the PSM Control Program manually, logon to the virtual machine in the usual way. The PSM Control Program can be started by simply entering its program name, as selected by the programmer at installation time. For example, if the programmer named the program ISQPSM, then simply enter this in response to a Ready prompt:


```

Ready; T=0.04/0.05 19:52:14
isqpsm
ISQMA0001I PSM Main Thread entered
ISQMA0101I Thread 3 started as CSERV
ISQMA0002I Waiting for Command Server thread to initialize
ISQCS0710I Command Server thread entered
ISQCS0205I Command Server valid FP addresses are 9.164.172.63 9.164.156.189 9.99.99.99 66.77
ISQMA0101I Thread 4 started as CNSERVER
ISQMA0101I Thread 5 started as LOGGER
ISQCN0710I Console Server thread entered
ISQCN0711I Console Server thread initialized
ISQCS0711I Command Server thread initialized      ISQLG0710I Logger thread entered
ISQLG0711I Logger thread initialized

```

Initially PSM messages are displayed on the VM console. These may be captured to a disk file by spooling the console file to a reader file. To initiate spooling enter a CMS SPOOL command; for example:

```
spool console start *
```

Then start the PSM, as above. When you wish to create a complete file in the reader, enter a command:

```
spool console close
```

Spooling continues, and you may use the above spool console close command to create successive reader files. When spooling is to stop, enter the command:

```
spool console stop
```

(This does *not* write the remaining records to a reader file.)

If the above commands are to be entered while the PSM Control Program is running, the *spool* commands should be prefixed with *cms*:

```
cms spool console close
```

To automate the process of starting the PSM Control Program when the PSM virtual machine starts, add the ISQPSM command to the PROFILE EXEC of the PSM virtual machine.

On occasions, you may wish to start the PSM Control Program with traces active for threads that are started automatically. Before issuing the command to start the PSM, enter a CMS command of the following form:

```
GLOBALV ISQTrace.comp ON
```

Where *comp* is the trace option as used in the ISQTRACE request. (Usually one of CSERV, TCPIP, LOGGER or CNSERVER). Enter one command for each trace required.

Stopping the PSM Control Program

To stop the PSM Control Program normally, first logon to the PSM virtual machine. Because the PSM virtual machine runs disconnected, the first action is to issue a BEGIN command to resume operation.

```

LOGON PSM
  ICH70001I PSM LAST ACCESS AT 17:08:54 ON MONDAY, OCTOBER 20, 2003
z/VM Version 4 Release 3.0, Service Level 0202 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES: 0038 RDR, 0003 PRT, NO PUN
RECONNECTED AT 20:25:10 CST WEDNESDAY 10/20/03

```

```
begin
```

Then enter the STOPALL request. The PSM control program should then close its threads.

After a short while (about ten seconds) the CMS prompt should be given:

```
stopall
ISQCN0402I Command <stopall> entered
HCPMFS057I PSM not receiving; not authorized
ISQCS0218I Request <stopall> done.
ISQCN0708I Console Server thread canceled
ISQCN0709I Console Server thread ending...
ISQCS0708I Command Server thread canceled
ISQLG0708I Logger thread canceled
ISQCS0709I Command Server thread ending...
ISQLG0709I Logger thread ending...
ISQMA0005I Main Thread ending (others may still be running)
Ready; T=0.09/0.10 20:30:59
```

Sometimes it is not possible to use STOPALL to stop the PSM. In these cases, enter the HX command (halt execution) followed by B (BEGIN), repeatedly until the Ready prompt is obtained:

```
hx
CMS
b
DMSRXS1419E EventSignal failed for event CONNSOCK; RC=8 Reason=108
Ready; T=0.14/0.14 20:36:31
```

The DMSRXS1419E message is normal

Restarting the PSM Control Program

After a PSM Control Program has stopped control returns to CMS and a CMS Ready prompt is issued to a logged on user. At this point it is advisable to enter a HX command to terminate any running threads.

The PSM Control Program can be restarted by entering its program name, for example, ISQPSM.

When the program restarts any trace setting that were set from the last execution are retained and are still in effect. If you wish to set off a trace before resuming execution enter the CMS command:

```
GLOBALV ISQTrace.comp OFF
```

Where *comp* is the trace options as used in the ISQTRACE request. Enter such a command for each trace that is to be set off.

Part 3. Using SA z/OS on Workstations

Chapter 11. Using the NetView Management Console for SA z/OS

The NetView Management Console (NMC) is a part of the NetView product and is used for monitoring network resources. As SA z/OS is based on NetView, it also exploits the graphical capabilities of the NMC.

You can monitor and control your whole enterprise with the NMC. It is also possible to monitor your enterprise from a single workstation (single point of control). This is valid for system operations as well as for processor operations. For general information about the NMC see *NetView Management Console User's Guide*. This chapter only describes the extensions that have been added for monitoring SA z/OS resources.

Introduction

The communication between SA z/OS and the NMC is maintained by the SA z/OS topology manager. An SA z/OS topology agent on each target system retrieves the enterprise data from the automation manager. The SA z/OS topology manager on the focal point provides the information into RODM. GMFHS takes the information from RODM and presents it in a graphical form on the NMC workstation. The following sketches the main tasks of the SA z/OS topology manager for system operations. For processor operations, the mode of operation is similar.

At initialization time, the SA z/OS topology manager retrieves SA z/OS information for one or more sysplexes from the respective automation managers. To do that, it contacts an SA z/OS topology agent that resides on a system within the sysplex to be monitored. The SA z/OS topology agent obtains the information from the automation manager of the sysplex and passes it to the SA z/OS topology manager.

The retrieved information includes all resources, for example, applications (type APL), application groups (type APG), systems (SYS), and groups of systems (GRP) with their mutual dependencies, and relationships. The SA z/OS topology manager stores these resources together with their dependency information within RODM.

The SA z/OS topology manager calls BLDVIEWS to build the views that are necessary for these resources to be displayed on the NMC. The definitions for the views must be supplied by the users.

This process is illustrated in Figure 125 on page 218. The arrows with broken lines represent requests, those with solid lines stand for delivery of information. This figure shows an idealized system configuration. Any constellation of systems is supported according to the abilities of SA z/OS. For example, all components can be installed on a single system.

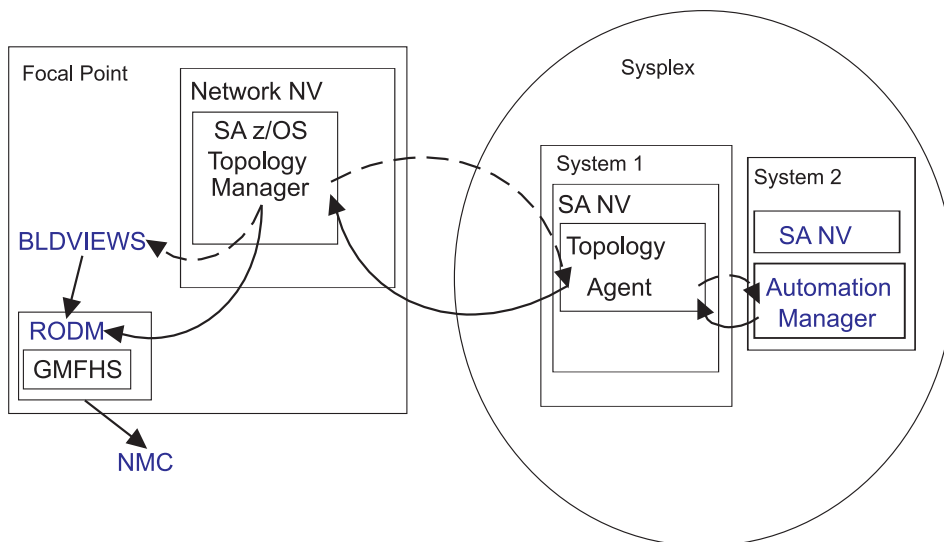


Figure 125. Overview

Note: In SA z/OS, a single system can be regarded as a special sysplex with one system only.

At runtime, the SA z/OS topology manager performs the following tasks:

- It updates the NMC whenever the status of a displayed resource changes.
- It dynamically displays events like WTORs or tape mount requests on the NMC.
- Users can issue SA z/OS commands from the NMC which will be routed by SA z/OS to the appropriate system for processing.
- It resynchronizes its resource structure for one or more sysplexes with the respective automation managers. When this happens, the views affected by eventual changes are rebuilt.

Starting the NMC

Before you start the NMC, make sure that the following conditions are satisfied:

- The NMC server must be up and running and must be communicating with NetView.
- In order to issue interactive commands (see “Interactive Commands” on page 227), you must have set up your NetView 3270 Management Console correctly. See *System Automation for z/OS Planning and Installation* for details.

After you have started the NMC, the **Sign On** panel is displayed. Enter the user ID and the password in the respective fields. The user ID must be identical to that of an active NetView host session. In the **Host Machine** field you must enter the address of the NMC server. If you do not know the correct address, contact your system administrator.

When you have signed on successfully, the **Tivoli NetView for z/OS Management Console** is displayed. The left part of that window contains the *Business Tree*. If you select a view from the *Business Tree*, it is displayed in the right part. The SA z/OS views are descendants of the **Networking** view as shown in the following figure.

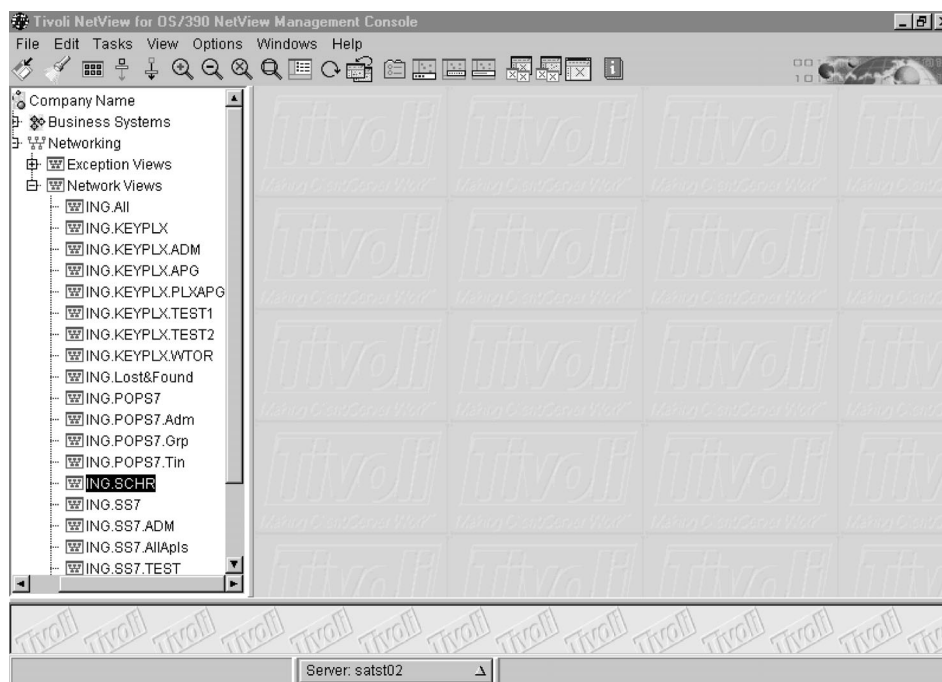


Figure 126. Tivoli NetView for z/OS Management Console

Monitoring SA z/OS on the NMC

On the NMC workstation, you can see the graphical visualization of what is automated in system operations and processor operations.

For system operations, the automation manager knows about the configuration of the sysplex and the resources running in it. The SA z/OS topology manager communicates with these automation managers.

For processor operations, it is the processor operations focal point which knows about the processor operations configuration and therefore, the SA z/OS topology manager communicates with it.

Besides the resource information retrieved from the automation managers, the SA z/OS topology manager maintains SA z/OS objects that have no counterpart within the automation manager. The following sections explain the representation of these two types of SA z/OS resources on the NMC.

Regular Resources

SA z/OS resources in RODM that also exist within the automation manager are called *regular*. For example, systems (SYS resource type), system groups (GRP), applications (APL), and application groups (APG) are regular resources. Figure 127 on page 220 shows a network view of regular resources.

Monitoring SA z/OS on the NMC



Figure 127. Network View for Regular Resources

The members of a group are represented on the NMC by subviews. For instance, if you double click on the application group `SS7.AOC7.APG.CICSGRPJSHP` of Figure 127, the following view will be displayed:

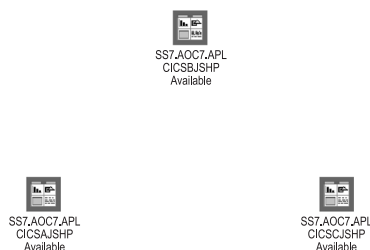


Figure 128. Subview with Applications of an Application Group

Minor Resources, Anchors and Tokens

The SA z/OS topology manager distinguishes between *major* and *minor* resources. Major resources, such as applications or systems, exist independently. Minor resources are subcomponents of applications (or MVS components) that cannot exist independently, but are always associated with a major resource. Typical examples for minor resources would be the spool of a JES application or transactions of a CICS application. Minor resources have their own automation flags so that you can, for example, switch on automation for an application, but switch it off for one of its subcomponents. This is one main reason for introducing the distinction between major and minor resources.

The minor resources that are defined in the policy database and known to SA z/OS are a special type of regular resources. They are retrieved by the SA z/OS topology manager from an automation manager (either during initialization of the SA z/OS topology manager or when the information in RODM must be resynchronized with the automation manager).

In addition to regular minor resources, the SA z/OS topology manager also knows *dynamic* minor resources. Dynamic minor resources are posted to the SA z/OS topology manager at runtime with the `INGPOST` command. You can also change the state of an existing dynamic resource with `INGPOST`. For more information on

the INGPOST command, see *System Automation for z/OS Programmer's Reference*. In contrast to regular resources, you can delete dynamic minor resources in RODM from the NMC.

To post a dynamic minor resource to the SA z/OS topology manager, you must specify the major resource to which this minor resource belongs. When the SA z/OS topology manager finds the major resource within RODM it connects the new minor resource with the major resource and displays it immediately in all (network) views that contain the major resource.

The major resource specified to INGPOST can be a regular resource or an *anchor*. Anchors are special objects that exist in RODM, but not in the automation manager. They play the role of major resources and serve two purposes:

- They are used to display dynamic minor resources for which either the specified major resource or the specified sysplex does not exist in RODM. See “‘Lost&Found’-Anchors” on page 222.
- Anchors are used to display and group minor resources that derive from events such as WTORS and tape mount requests.

Anchors and Events

The SA z/OS topology manager is able to represent events such as WTORS or tape mount requests on the NMC. The events are posted to it with the INGPOST command. The SA z/OS topology manager treats them as dynamic minor resources; these resources are called *event resources*. Event resources are connected to an anchor as their major resource. All events connected to one anchor belong to the same event type. Thus there are, for example, anchors for WTORS, and anchors for tape mount requests.

If specified in the Customization Dialog the SA z/OS topology manager creates one anchor per sysplex for several event types, for example for WTORS and tape mount requests. Whenever an event of such a type is posted to the SA z/OS topology manager, it is attached to the appropriate anchor of the respective sysplex, and all views that include this anchor are immediately updated.

Thus, when a WTOR originating from system AOC7 is posted to the SA z/OS topology manager and AOC7 belongs to sysplex SS7, this WTOR is attached as a minor resource to the (generated) WTOR anchor of sysplex SS7 and dynamically added to any view that includes the WTOR anchor of SS7.

Besides this you can define your own anchors for any events in the customization dialogs. You must post these events to the SA z/OS topology manager with the INGPOST command.

Figure 129 on page 222 shows a view that contains WTOR, TAPE, and SYSPLEX anchors for the sysplex SS7 with event resources attached to some of them. It also shows the HEARTBEAT anchor that is automatically created.

Monitoring SA z/OS on the NMC

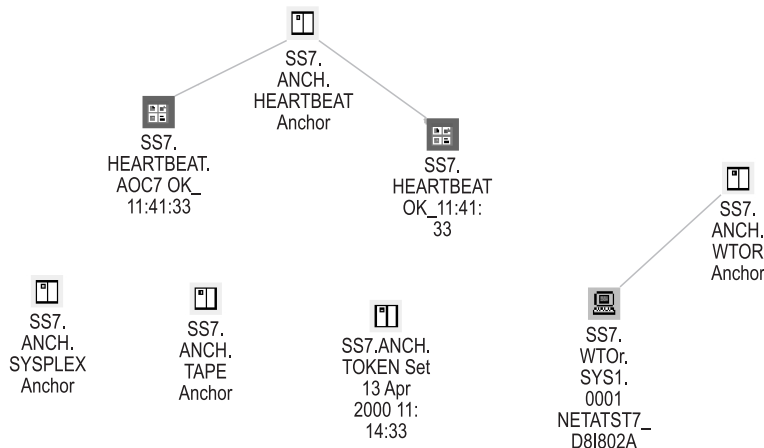


Figure 129. Anchor View

The anchors and their minor resources have the following function:

Heartbeat

These resources represent the connection between the SA z/OS topology manager and the sysplex. Usually, two minor resources are attached to the heartbeat anchor. One icon represents the heartbeat for the sysplex (the right one in Figure 129), the other is the heartbeat for the system that is currently being used to monitor the sysplex (the left one in Figure 129).

For more information on heartbeats, see *System Automation for z/OS Defining Automation Policy*.

Note: When the communication between the SA z/OS topology manager and the target system(s) fails, icons are created to represent the status: you will find a heartbeat icon with a corresponding label, such as *APLX.HEARTBEAT.NOTARGET* and a sysplex group labelled for example, *APLX.GRP.NoTarget NoCommunication*. The heartbeat icon is there to represent the state. On the sysplex group you can initiate an *INGTOPO REBUILD <system>* or an *INGTOPO SWEEPSYS <system>*.

WTOR

Write To Operator with Reply (WTOR) messages are z/OS messages that inform you of various conditions and require a reply of the operator. In Figure 129, one WTOR is attached to the WTOR anchor.

In the label of this WTOR, 'WTOR' is the major resource name, and 'SYS1.0001' the minor resource name. The additional information is the message content.

By defining a view that contains the WTOR anchors of all sysplexes in the enterprise, you can monitor all WTORS at a single glance. For defining views, see "Creating Views" on page 230.

Tape mount request

When a tape mount request for sysplex SS7 is posted to the SA z/OS topology manager, it is attached to the tape mount request anchor of this sysplex.

'Lost&Found'-Anchors

To post a new minor resource to the SA z/OS topology manager, you must specify the major resource to which this minor resource belongs. When the SA z/OS

topology manager cannot find the major resource in RODM, it attaches the minor resource to one of several special anchors that serve to display such unconnected minor resources.

There is one such anchor at sysplex level, which collects minor resources whose major resource does not exist in RODM, but whose sysplex is known to the SA z/OS topology manager. If even the sysplex is unknown, the SA z/OS topology manager attaches the minor resource to an anchor with the prefix 'Lost&Found'. There are 'Lost&Found'-anchors for WTORs, tape mount requests, heartbeats, and for sysplex related resources.

Figure 130 displays a view that comprises the 'Lost&Found'-anchors.



Figure 130. View with 'Lost&Found'-Anchors

Tokens

A token is an object, which is generated automatically by the SA z/OS topology manager. The date and time when the SA z/OS topology manager was initialized is appended to the name of the token, In the *Data1* field of the resource properties, you can find the original ACF token or the processor operations token of the monitored system. This can be used for diagnostic purposes.

Properties of SA z/OS Objects on the NMC Workstation

All resources are represented by icons. Dependencies are displayed as lines; no difference is made between start and stop dependencies. Association of a minor resource to a major resource is also represented through a line. The state of the resources is indicated by the color of the icon. The colors are mapped to the compound states of the resource as follows:

GREEN	Satisfactory
LIME	InAutomation
CYAN	Awaiting
WHITE	Inhibited
YELLOW	Degraded
PINK	Denied
RED	Problem
GREY	Status not known (NMC specific)
BLACK	Resource deleted in RODM (NMC specific)

For more information on the compound status, see "Compound Status" on page 37.

The labels of the icons consist of three (four) parts, namely

1. The sysplex name as defined in the INGTOPOF file.

For the INGTOPOF file, see *System Automation for z/OS Planning and Installation*.

Monitoring SA z/OS on the NMC

2. A derivative of the major resource name.

This part of the label differs from the resource name within the automation manager in that the order of its components is reversed and the slashes are replaced by dots (for example, the standard resource name JES2/APL/SYS1 becomes SYS1.APL.JES2). The reason for this reversal is explained in “Names” on page 231.

3. Possibly a minor resource name.

The minor resource name of the JES2 spool in Figure 127 on page 220 is SPOOL. When a minor resource name contains more than one part, the separator slashes are also replaced by dots, but no reversal of the components takes place (for example, TRAN/ABC123 would become TRAN.ABC123).

4. Additional information.

With major resources, this additional information is essentially the compound status. In most cases, it is supplemented by a suffix that indicates whether the status refers to starting or stopping the resource. Thus there is an INHIBITED_START state and an INHIBITED_STOP state. The SATISFACTORY compound state is replaced by the pair AVAILABLE and UNAVAILABLE.

For the minor resource in Figure 127 on page 220, the additional information is ‘OK’.

For more information on the labels, see “Names” on page 231.

If you have the appropriate CPE profile setup for your user ID, you can view and possibly edit the **Resource Properties** of the displayed resources (see also *NetView Management Console User’s Guide*).

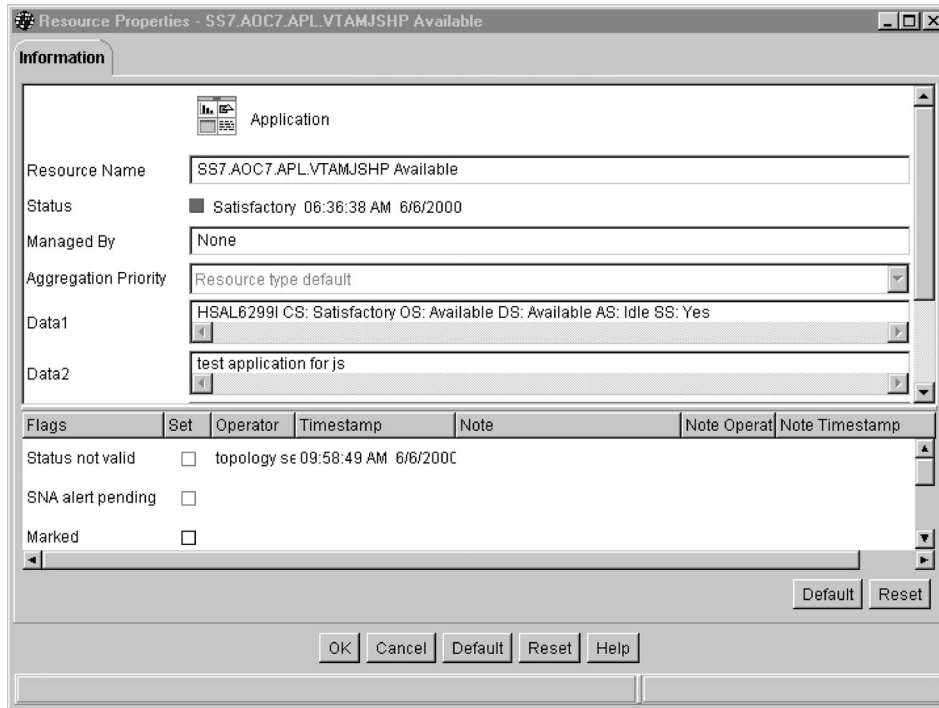


Figure 131. Resource Properties Window

For regular major resources, the **Data 1** field contains detailed status information, and the **Data 2** field can contain a comment, for example from an operator. For dynamic minor resources (see “Minor Resources, Anchors and Tokens” on page 220

220), the **Data 1** field corresponds to the message parameter of the INGPOST command, **Data 2** corresponds to the userdata parameter of the INGPOST command, and **Data 3** can be changed by using the D3 parameter of the INGPOST command; see *System Automation for z/OS Programmer's Reference* for more information on INGPOST. **Data 4** is not currently used.

SA z/OS information is presented on the NMC in *network* views or in *exception* views. Network views represent the topology of resources, that is, grouping of resources and their connections. Figure 127 on page 220 is an example for a network view. Exception views display resources that have certain properties in common.

Dynamic Behavior of the SA z/OS Topology Manager

In a running SA z/OS enterprise, you can imagine scenarios where new systems are IPLed while the SA z/OS topology manager is already running, or new systems are added to a sysplex and so on. The following sections describe how the SA z/OS topology manager behaves in these events.

SA z/OS Topology Manager Initialization: INGTOPO INIT

If the systems are set up accordingly, SA z/OS automatically issues the INGTOPO INIT command after it received the completion code of the start of the NetView MultiSystem Manager. If this is not the case, you have to start the SA z/OS topology manager manually by issuing the INGTOPO INIT command.

With the INGTOPO INIT call, the SA z/OS topology manager queries the INGTOPOF file and contacts each of the specified target systems (the specified sysplexes or the processor operations focal point). For the contacted target systems, it creates the according objects in RODM.

System Detection

When a system on which SA z/OS is installed becomes active, it sends a signal to the SA z/OS topology manager. The SA z/OS topology manager, which is already running, checks whether the system belongs to a sysplex or a processor operations focal point it knows (i.e., that is defined in the INGTOPOF file) and whether this sysplex or processor operations focal point was incommunicative until now. If both conditions are fulfilled, the SA z/OS topology manager performs the initialization process for this sysplex or processor operations focal point. In all other cases the SA z/OS topology manager ignores the signal. Note that these signals never reach any operator when the SA z/OS topology manager is not present.

The Heartbeat Function

When the SA z/OS topology manager makes contact with a system that can act as a representative of a sysplex, it launches a *heartbeat* timer on that system and a *deadman's* timer for that sysplex on the focal point system. The heartbeat timer sends a generic alert to the focal point system at a regular interval to indicate that the target system is still in good health. This alert resets the deadman's timer on the focal point. If the alert does not arrive before the deadman's timer expires, the SA z/OS topology manager assumes that the current representative cannot respond. The processing in such a case is described in "Takeover when a System Goes Down" on page 226.

Takeover when a System Goes Down

If the SA z/OS topology manager needs to assume that the sysplex representative cannot respond, it tries to find another one according to the specified systems in the in the SYSPLEX statement in the INGTOPOF file. Therefore the sequence of the specified SYSPLEX and PROCOPS statements and the sequence of the specified systems is relevant. If it finds a new representative, communication continues with the new one and the representation on the NMC continues. If it finds none, it considers the communication with the respective sysplex interrupted; all resources of the sysplex assume the color representation of the status *Unknown*.

Issuing Commands from the NMC

In order to issue a command against a resource, click on this resource with the right mouse button to open a context menu. This menu contains only those system operations or processor operations commands that are applicable to the respective resource. Some SA z/OS commands are not provided because they cannot be reasonably connected to a resource.

When the state of the resource is affected by the command, the state information (event component of the label, color of the icon) is actualized on the NMC.

The following sections describe the different command types.

Direct Commands

Some commands, for example INGINFO, need no user specified parameters. They are immediately executed after having been selected in a command submenu. Their output is displayed in the log of the NMC.

Dialog Commands

Most SA z/OS commands accept or need parameters that you must specify. When you select such a command from a submenu, a window will open where you can enter the appropriate values. Their output is displayed in the log of the NMC.

If you select, for example, the command INGTHRES for the resource SS7.AOC7.APL.VTAMJSHP in the view of Figure 127 on page 220, the appropriate parameter window is opened as shown in Figure 132 on page 227.

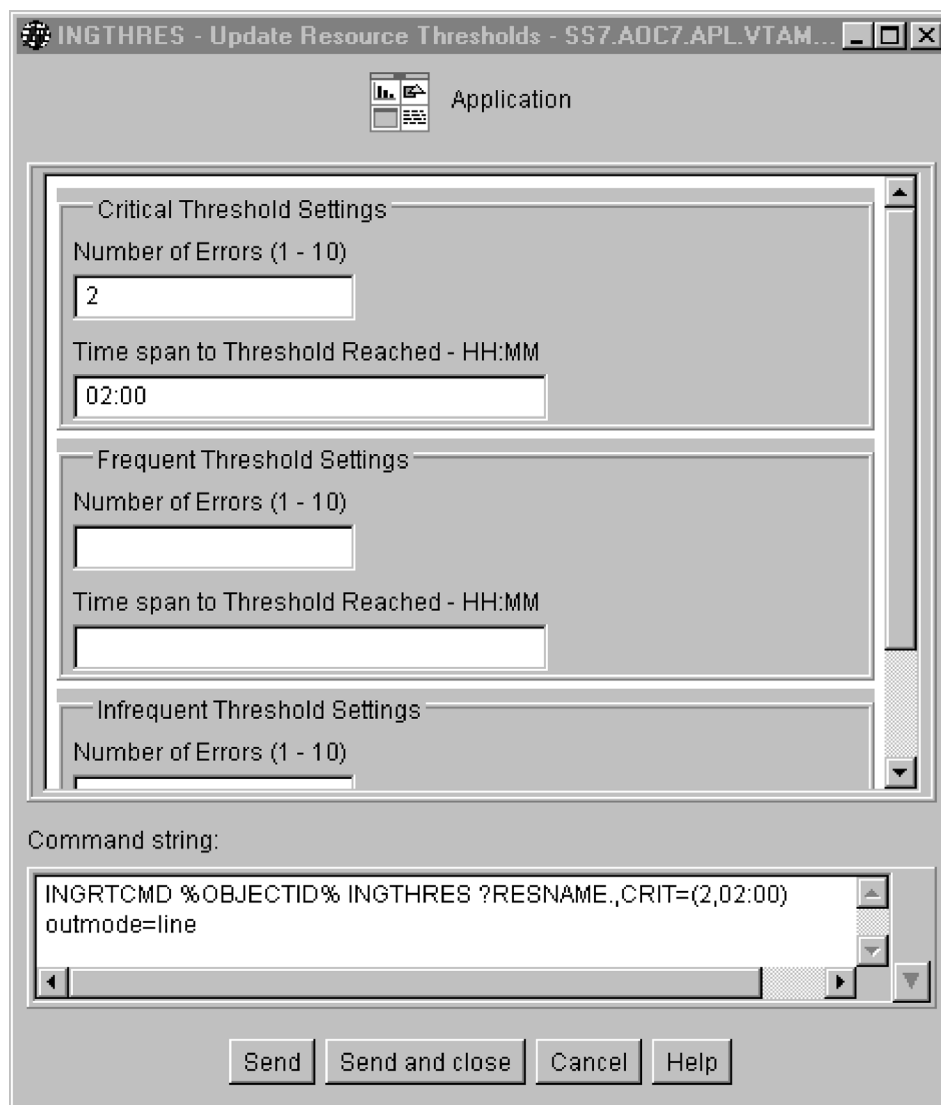


Figure 132. Parameter Window for INGTHRES

All parameter windows consist of two parts. The upper part contains the fields where you can enter the parameter values. The lower part displays the operator command that is generated from your input in the upper part. All you have to do is to enter the required values or choose the required option in the parameter fields (which has already been done in Figure 132) and issue the command by clicking on one of the two **send** buttons.

You can edit the command string in the **Command string** box at the bottom of the window. However, this is not necessary and may even lead to errors if you do not have a thorough understanding of the INGRTCMD command. “The INGRTCMD Command” on page 229 contains a brief summary of the INGRTCMD mechanism.

Interactive Commands

There are commands that require a continued dialog before they can be processed. Such commands are routed to a NetView 3270 Management Console and are processed there; this presupposes that you have set up the 3270 console correctly.

The INGREQ STOP command provides an example:

Issuing Commands from the NMC

After clicking on the resource SS7.AOC7.APL.VTAMJSHP in Figure 127 on page 220 and selecting the entry **SA Operations Commands**, a submenu is opened which includes the **INGREQ Stop** entry. If you select this command, the request is forwarded to a NetView 3270 console, which would look as follows:



Figure 133. NetView Console with INGREQ Panel

After pressing ENTER you are presented with a list of the resources affected by the command (SS7.AOC7.APL.VTAMJSHP and SS7.AOC7.APG.CICSGRPJSHP) . If you confirm the request the resources are stopped. The view of Figure 127 on page 220 is actualized, and now looks as follows:



Figure 134. Actualized View

Note the modified event components in the label of the resources SS7.AOC7.APL.VTAMJSHP and SS7.AOC7.APG.CICSGRPJSHP.

Note: Depending on your setup, at the initial start of the NetView 3270 Management Console, triggered by an SA z/OS command that needs to be transferred to the NetView 3270 Management Console, you might be confronted with messages, or even problems (e.g. failing logon), that are not related to the original SA z/OS command. Handle these messages or problems as appropriate and retry the command.

The INGRTCMD Command

When you select a resource and then choose an operator command, a template of that command is generated. In this template, a place holder value is specified for all parameters of the command to which an attribute of the selected resource can be assigned as its value. The place holder points to this attribute. The place holders all have the format *?name..*. Thus, for example, *?RSC.* refers to the resource name (which would be VTAMJSHP/APL/AOC7 in our example) of the object you clicked on.

Before the command is forwarded to the appropriate target system, these place holders must be replaced by real values. This is done by the command INGRTCMD. In order to be able to substitute the placeholders, INGRTCMD must be passed the ID of the object whose attributes are to supply the parameter values, so that the format of the command is as follows:

```
INGRTCMD object_id command_string
```

In the parameter window, the *object_id* is specified in the form %OBJECTID%, which is a reference to the RODM object ID of the selected resource. The command string displayed in Figure 132 on page 227 would thus be converted by INGRTCMD into

```
INGTHRES VTAMJSHP/APL/AOC7,CRIT=(2,02:00) outmode=line
```

and would be routed to the system representing the sysplex SS7.

Creating Views

The RODM objects representing SA z/OS resources must be included in a view in order to be displayed on the NMC. The definition of these views is up to you. The SA z/OS topology manager, however, offers some support for executing view definitions and updating the views whenever necessary. In order to exploit this feature, you must specify your view definitions in the format required by the NetView BLDVIEWS tool. Some familiarity with this tool and with the terminology and class structure of RODM (see *Tivoli NetView for z/OS Resource Object Data Manager and GMFHS Programmer's Guide*) will be assumed in the following description.

The files containing your BLDVIEWS view definitions can be introduced to the SA z/OS topology manager in the INGTOPOF configuration file (for information on the INGTOPOF file, see *System Automation for z/OS Planning and Installation*). When the SA z/OS topology manager is started, it will, after constructing the SA z/OS objects, create the views in RODM by calling BLDVIEWS for all the definition files that it finds in the INGTOPOF file. BLDVIEWS will create a view object in RODM for every definition and connect every resource that meets the selection criteria specified in the definition with this view object.

At runtime, it will be necessary to rebuild the views for certain sysplexes when their SA z/OS objects in RODM are not, or no longer, current. In such a case, it is desirable that the views be rebuilt only for those sysplexes whose SA z/OS structure has in fact changed. For this reason, the definition files are associated in the INGTOPOF file with sysplexes. Thus, when defining a sysplex specific view, be sure to put it into a file that is associated in INGTOPOF with this sysplex. The same consideration applies to sets of target processors. For more information, see "Network Views and Exception Views."

Note: You can also call the BLDVIEWS tool independently of the SA z/OS topology manager to update views or create further views. However, these will not be automatically updated when new major resources are created in RODM.

Network Views and Exception Views

NetView provides different types of views. For displaying SA z/OS resources, only *network views* and *exception views* are used.

Network views are intended to display the structure of a network. Therefore, all relationships between resources are displayed on the NMC. Furthermore, network views are static with regard to their structure; unless the link between a view element and the view object is removed or a new link is established, they always contain the same elements.

Exception views, on the other hand, display resources that have certain properties in common, for example, all resources that are in an exception state and marked by an operator. The relationships are not displayed in an exception view. Exception views are dynamic; although the underlying structure does not change, new elements are included in the view when a resource acquires the appropriate combination of properties (for example, when an operator marks a resource that is in an exception state), and elements are removed when they change any of their relevant properties (for example, when the resource is unmarked again).

A BLDVIEWS definition of a view typically consists of two parts. With the VIEW (for network views) or EVIEW (for exception views) statement you specify the type and properties of the view.

The second part of the definition specifies the resources that are to appear in the view. All SA z/OS resources in RODM are Non-SNA (GMFHS Managed Real) resources. Thus, you can use either the **MyName** or the **DisplayResourceName** field as a selection criterion for the Non-SNA resources to be included in the view. Note that you can specify more than one selection criterion.

Following are a typical network view and a typical exception view definition. First the network view:

```
VIEW=ING.SS7.APPL,ANNOTATION='Sysplex SS7'
NONSNA='SS7.AOC7.*',QUERYFIELD=DRN
```

The NONSNA statement must be read as "Include in the view all Non-SNA resources whose **DisplayResourceName** field contains a value beginning with SS7.AOC7.." For more information on view definitions for the BLDVIEWS tool see the online help which is shipped with NetView.

Now a typical exception view:

```
EVIEW=ING.SS7.EXC,ANNOTATION='Exceptions SS7',DSF=NOXCPT,MF=OFF
NONSNA='SS7.AOC7.APL.*',QUERYFIELD=DRN
```

Note the interplay between the selection criterion of the NONSNA statement and the filter parameters of the EVIEW statement. In contrast to the network view, the NONSNA statement for an exception view only specifies the *candidates* for appearance in the view. Which of these candidates are in fact included depends on the filter parameters. In the example definition, the exception state filter (DSF parameter) and the mark filter (MF parameter) are set so that only those candidate resources are displayed that are in an exception state and marked by an operator.

In order to be able to specify the selection criteria correctly, you must know how the names of the SA z/OS resources in the RODM **MyName** and **DisplayResourceName** fields are constructed. This is explained in the following section.

Names

Every SA z/OS resource in RODM has two names, an *object name* (RODM **MyName** field) and a *display name* (RODM **DisplayResourceName** field). The object name identifies the object uniquely within RODM, the display name appears as the label of the icon that represents the object on the NMC. When defining a view to BLDVIEWS, you can use either of these names as a selection criterion for the objects to be included in the view.

Since the names for system operations objects and for processor operations objects are rather different, they are described separately.

Names for System Operations Objects

The object names for system operations objects have the following format:

```
sysplex_name.{name/type/system | name/type | ANCHOR/MJR/anchor_name}
[minor_res_name]
```

Here *sysplex_name* is the name of a sysplex as defined by a SYSPLEX statement in the INGTOPOF file.

Creating Views

The display names have the following format:

```
| sysplex_name.{system.type.name | type.name | anchor_name.MJR.ANCHOR}  
| [.minor_res_name] [ev_summ]
```

Here again, *sysplex_name* is the name of a sysplex as defined in the INGTOPOF file. Note that the order of the component names for major resources is reversed in the display name. This is done to provide more flexibility when specifying filter criteria for views. Note further that slashes (/) are replaced by dots in the names of minor resources. *ev_summ* stands, in the case of a regular major resource, for the current compound status of the resource. With minor resources it stands for various information, for example for the message content of a WTOR or the timestamp of a heart beat.

The following are some examples for both name types:

1. Application TSO on system AOC7 in sysplex SS7:

Object Name:

SS7.TSO/APL/AOC7

Display Name:

SS7.AOC7.APL.TSO

2. Regular minor resource SPOOL of application JES2JSHP on AOC7 in sysplex SS7 in state OK:

Object Name:

SS7.JES2JSHP/APL/AOC7.SPOOL

Display Name:

SS7.AOC7.APL.JES2.SPOOL OK

3. WTOR outstanding on system AOC7 in sysplex SS7:

Object Name:

SS7.ANCHOR/MJR/WTOR.AOC7/0007

Display Name:

SS7.WTOR.MJR.ANCHOR.AOC7.0007 NETCTST7_DSI802A

Note: The display name can be customized. For further information, refer to "Appendix B, Syntax for INGTOPOF File" in *System Automation for z/OS Planning and Installation*.

Names for Processor Operations Objects

The object names for processor operations objects have the following format:

```
procops_name.name/type[/focal_point_system]/1st_level_min[/2nd_level_min  
[/3rd_level_min]]
```

Here *procops_name* is the name of a focal point configuration as defined by a PROCOPS statement in the INGTOPOF file.

The display names have the following format:

```
procops_name.[focal_point_system.]type.name.1st_level_min[.2nd_level_min  
[.3rd_level_min]] [ev_summ]
```

Here again, *procops_name* is the name of a focal point configuration as defined in the INGTOPOF file. As with the system operations names, the order of the major

resource component names is reversed, and slashes (/) are replaced by dots. *ev_summ* stands for the current processor operations state of the resource.

SA z/OS Topology Manager Commands

There are two commands through which you can communicate with the SA z/OS topology manager, namely INGTOPO and INGPOST. For more information on these commands, also refer to *System Automation for z/OS Programmer's Reference* and *System Automation for z/OS Operator's Commands*.

INGTOPO Command

This command gathers SA z/OS information about one or more sysplexes (systems) or sets of target processors from the respective automation manager(s) or focal point processor(s) and updates the SA z/OS structure for these sysplexes (systems) within RODM accordingly. You must call it with a parameter that specifies its mode of operations. The following parameters are valid:

INIT[*task_name*]

In INIT mode, INGTOPO sets up the sysplex (processor operations focal point) for monitoring and builds or updates the SA z/OS objects in RODM in accordance with the automation managers of all sysplexes (systems) that are defined in the INGTOPOF file. INIT is the mode which you must specify at the very first call of INGTOPO. In this case, you must also specify the name of the autotask under which the SA z/OS topology manager runs.

You may call INGTOPO with INIT any time. INGTOPO will then either rebuild the SA z/OS objects from scratch or just perform the necessary update operations.

REBUILD *sysplex_name_or_procops_name*

In REBUILD mode, INGTOPO removes all SA z/OS objects in RODM for the specified sysplex (or processor operations focal point) and rebuilds them from scratch. You can use this call to rebuild the SA z/OS information for a sysplex (or processor operations focal point) when the RODM data for this sysplex (or processor operations focal point) appear to be incomplete or damaged.

RESYNC *sysplex_name_or_procops_name*

In this mode, INGTOPO resynchronizes the SA z/OS information in RODM with the specified sysplex (or processor operations focal point). The INGTOPO command with the RESYNC option is typically initiated automatically by the target system. This may only involve a status update, but can also mean a complete rebuild of the SA z/OS information, if this is necessary.

SWEEPSYS *sysplex_name_or_procops_name*

In this mode, INGTOPO removes all SA z/OS objects in RODM for the specified sysplex or processor operations focal point and stops all monitoring services.

SWEEP

In this mode, INGTOPO removes all SA z/OS objects in RODM for all sysplexes and all processor operations focal point and stops all monitoring services.

For more information, see *System Automation for z/OS Operator's Commands*.

INGPOST Command

This command is normally invoked from within a program. It is used to dynamically display subsystems or other events. For example, if you have a CICS running, and one of its transaction runs into problems, you could specify an INGPOST which links a corresponding minor resource of this transaction to the CICS major resource, specifying the event and indicating its status by a corresponding color.

The purpose of INGPOST is to enhance problem management. You can also use INGPOST to visualize any other alerts or events and to create or manipulate minor resources.

For more information on INGPOST, see *System Automation for z/OS Programmer's Reference*.

Processor Operations on the NMC

Processor operations deals with controlling operations of the processor hardware. One or more of these processors are controlled from a focal point. There can be more than one focal point. It is these focal points that the SA z/OS topology manager contacts for gathering information. A focal point with its associated target processors is called a processor operations focal point configuration or, if no confusion is likely to arise, simply a *focal point configuration*.

Processor operations is independent from system operations. Accordingly, when the same enterprise component (for example, a system) is integrated into processor operations and also into system operations, the SA z/OS topology manager creates two representations of it in RODM. For these reasons it will usually be advisable to keep the processor operations objects apart from the system operations objects in the NMC views. See "Creating Views" on page 230 for defining views.

The following figure shows a network view of a processor operations configuration.

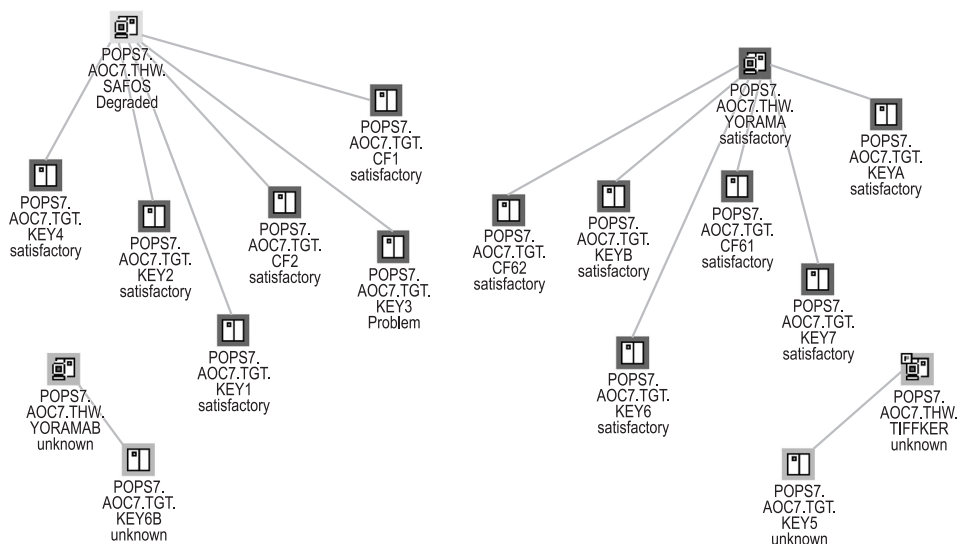


Figure 135. Processor Operations View

The SA z/OS topology manager knows four kinds of processor operations objects which can be distinguished by the type component of their labels. This label is constructed as follows:

1. The name of a focal point configuration as defined in the INGTOPOF file.
For the INGTOPOF file, see *System Automation for z/OS Planning and Installation*.

2. The resource name.

This consists of the name itself, the type identifier, and optionally a system name.

There are four resource types, whose names are derived (with one exception) from the corresponding processor operations type names by omitting the prefix ISQ. The names have the following meanings:

TGT This characterizes a target system.

THW This characterizes a target processor.

PGR This characterizes a group of target systems.

Note: This type identifier is different from the second part of the corresponding processor operations type name, which is ISQGRP. The reason for this deviation is that GRP is already used within system operations.

SBG This characterizes a subgroup of target systems.

3. A minor resource name consisting of one, two or three parts. The parts are separated by dots.

The states of processor operations objects are interpreted in terms of NMC states. You can view the complete processor operations status in the **Data 2** field of the **Resource Properties** window.

Monitoring Sysplex Resources

The SA z/OS topology manager detects and provides objects that represent sysplex resources (for example, coupling facilities and couple data sets). Also sysplex resource specific commands are provided so that you can issue them in the same way as described in "Issuing Commands from the NMC" on page 226.

Problem Management

Problem management facilities offered by the NMC workstation consist of the following aspects described in the subsequent sections:

- "Exception Views"
- "Marked and Unmarked Resources" on page 236
- "Updating Operator Data" on page 236
- "Problem Management of Minor Resources" on page 237

These sections are an outline of suggestions how you can use the combination of these facilities to establish an efficient problem management.

Exception Views

Exception views are typically used for problem management. It is recommended that you define your exception views in pairs, where the difference between the two views is that one of them contains only unmarked resources, and the other only marked resources. For example, the marked resources could be those being

Problem Management

worked upon by an operator, the unmarked ones are not yet handled. The exception views are actualized dynamically; if you mark a resource within a view that contains only unmarked resources, the resource is immediately removed from that view and added to the corresponding view with marked resources.

The following view, for example, shows one of the resources also occurring in Figure 127 on page 220 that is in an exception state.



Figure 136. Exception View

For more information on network and exception views, see “Network Views and Exception Views” on page 230.

Marked and Unmarked Resources

The feature of marked and unmarked resource as introduced in “Exception Views” on page 235 is provided for every resource via the resources command menu. The NMC workstation just lets you mark resources. Marked resources are displayed with a flag. You define within your enterprise what the meaning of the mark should be. So for example, you can distinguish resources in different states of attendance. Marked and unmarked resources dynamically change within the appropriately defined views according to your marking actions.

| Markings processed through the SA z/OS marking commands (offered in the
| popup menu) can be persistently stored by SA z/OS. This is the advantage over
| the standard marking function as provided by native NMC.

Updating Operator Data

You can write messages into the **Data 2** field of the **Resource Properties** window from the NMC. This can be used for problem management. For example, when you mark a resource that is in an exception state in order to indicate that you are working on the problem you can add in this way a comment for your colleagues.

You call this function by selecting **SA Icon Commands** and then **Update Operator Data**. A window appears where you can enter your comment. This will immediately be written into the **Data 2** field of the selected resource.

| The advantage of this function, as compared to the standard function for
| information exchange between operators in the lower part of the **Resource**
| **Properties** window is that the comments can be persistently stored.

Problem Management of Minor Resources

Minor resources inherit the view properties of the related major resources. Thus they appear in the same views as the major resource.

For example, if a major resource could appear in an exception view, and now a minor resource of this major resource runs into a problem, then this minor resource will appear in the exception view (independently of whether the major resource is visible). However, as minor resources are bound to a major resource, they can only appear in views where you would expect the major resource.

You can apply problem management on minor resources just as you would do for major resources; that is, you can mark and unmark minor resources and you can also append a comment to them as described in “Updating Operator Data” on page 236. Additionally, in contrast to major resources, you can delete minor resources.

Problem Management

Part 4. TEC Notification by SA z/OS

Chapter 12. Using TEC Notification by SA z/OS

This chapter describes how to use TEC Notification by SA z/OS. After an introduction, “SA z/OS Messages, Alerts and Tivoli Events” on page 242 describes how SA z/OS messages and NetView alerts are mapped to Tivoli events. “Operation of TEC Notification by SA z/OS” on page 251 contains the information required to monitor Tivoli events from the Tivoli desktop. You find information on how to install TEC Notification by SA z/OS in *System Automation for z/OS Planning and Installation*.

Understanding the Use of TEC Notification by SA z/OS

SA z/OS is an availability management product and covers several aspects of automation and enterprise monitoring of z/OS for a single system as well as in a sysplex environment.

However, automation may fail and in such cases, messages or alerts indicate that administrator intervention is required. In those cases, SA z/OS sends out messages to the NetView operators and/or the system operator. The system operator is responsible for handling all events reported by SA z/OS from a single point of control for systems automation.

It may be desirable, however, to run SA z/OS integrated with Tivoli, so that events reported by SA z/OS as well as other events reported in a distributed heterogeneous environment are received at and handled from a single point of control. So SA z/OS is enhanced in a way that SA z/OS notifies the Tivoli Enterprise Console® (TEC) in cases when SA z/OS issues messages and alerts indicating critical situations. Such messages and alerts are forwarded to the TEC event server.

Thus, a Tivoli administrator running the Tivoli Enterprise Console may become aware of this event. The Tivoli administrator may also be notified when such an event is received by running an appropriate TEC rule which for example triggers a Tivoli notice to be sent to the Tivoli administrator’s notice board.

You may also correlate such an event with other events from the distributed heterogeneous environment so that the coincidence of correlated events may trigger a certain action.

Thus, a Tivoli administrator can use the Tivoli Enterprise Console as a single point of control for monitoring and handling events from both the distributed environment and SA z/OS.

This function of SA z/OS is called *TEC Notification by SA z/OS*. This notification is taking advantage of the *Tivoli Event/Automation Service*. In this environment, the source of notification is of no importance. All events occurring in your enterprise, originating from Tivoli NetView, SA z/OS, etc. are handled from the Tivoli Enterprise Console, which is the single point of control for Tivoli administrators.

The *Event/Automation Service* allows you to effectively manage both distributed and z/OS events from a single interface on a single console. You can exploit the event correlation and automation capabilities of the Tivoli Enterprise Console for

Understanding the Use of TEC Notification by SA z/OS

all events in the enterprise. You can find more information on the Event/Automation Service in *Tivoli NetView for z/OS Customization Guide*.

A Note on Terms:

A selection of SA z/OS messages is mapped to Tivoli events. In contrast to traditional Tivoli events, these new events mapped from SA z/OS messages will be called **SA z/OS events** in this documentation.

Introduction of TEC Notification by SA z/OS

The SA z/OS notification may notify Tivoli Enterprise Console (TEC) about an automation problem on z/OS by sending an event to the TEC event server.

For this purpose, on z/OS systems, messages or alerts are transformed into Tivoli events and sent to the *TEC event server* which is running on a Tivoli managed node in your network.

These events in turn may cause a notification of a Tivoli administrator on the TEC, may be correlated with other events on the TEC event server, or may result in opening a trouble ticket for example, dependent on what you programmed at the TEC event server.

The TEC event server is a notification target for SA z/OS. Note that not all messages, but only messages indicating critical situations and alerts are forwarded as TEC events to the TEC event server using the respective NetView Event/Automation Service Adapter.

A Tivoli administrator who wants to deal with a problem indicated by an event forwarded to the TEC event server by SA z/OS needs access to the affected z/OS system. For this purpose, you may use the Tivoli NetView 3270 Management Console. With TEC Notification by SA z/OS, the TEC administrator may log on to the NetView operator console by starting the NetView 3270 Management Console from the TEC console by executing a task. See “How TEC Notification by SA z/OS Works” on page 253 for a description on how to achieve that.

Note: Forwarding of SA z/OS messages to TEC will not start until SA z/OS and the Event/Automation Service are up and running. SA z/OS messages issued during SA z/OS startup will not be forwarded to TEC.

SA z/OS Messages, Alerts and Tivoli Events

The complete set of SA z/OS messages is not mapped to Tivoli events, but only an appropriate selection is handled by TEC Notification by SA z/OS. Table 10 on page 243 shows which messages are mapped to Tivoli events. Table 11 on page 246 shows a list of available Tivoli events mapped from SA z/OS messages and how they are integrated into the existing Tivoli event class structure. Table 12 on page 249 shows which alerts are mapped to Tivoli events.

For details on the format of message mapping or alert mapping see the respective shipped files in hlq.SINGSAMP: *INGMFMT* for message mapping and *INGACDS* for alert mapping.

Section “TEC Rules for SA z/OS Events” on page 249 describes the correlation rules between certain SA z/OS events defined by TEC Notification by SA z/OS.

Mapping SA z/OS Messages to Tivoli Events

In this section:

- you find information about the Tivoli event class structure,
- Table 10 shows how SA z/OS messages are mapped to Tivoli events (SA z/OS events)
- Table 11 on page 246 shows for all involved SA z/OS events, how SA z/OS messages are mapped to them.

For more information about the meaning of SA z/OS messages, refer to *System Automation for z/OS Messages and Codes*.

Event classes are defined hierarchically, with child classes inheriting attribute value defaults from the parent. The SA z/OS event classes follow a simple hierarchy.

All SA z/OS events are instances of event classes which inherit from *NV390MSG_SA* which in turn inherits from event class *NV390MSG_Event*, the top level class of all event classes. All event classes defined for the NetView message adapter inherit from this class. This inheritance structure is outlined in Figure 137 on page 246.

The following attributes are used in event group filters. They are set by the NetView z/OS Message Adapter and overridden by TEC Notification by SA z/OS:

sub_source

All events originating from SA z/OS messages are issued by SA z/OS system operations. This is indicated by `sub_sources="SA0S390_SysOps"`.

The following attributes are new with *NV390MSG_SA* and common for all SA z/OS events:

date_msg

Gives the date and time when the message is generated. This may differ from the date and time the event to which it is mapped is generated - see attribute **date**. The format is "MMM dd, hh:mm:ss" as in date and date_reception, for example, **Nov 03, 18:34:27**.

system_id

Is a 1-8 character identifier of the z/OS system image on which the message was issued.

sysplex_id

Is a 1-8 character identifier of the sysplex from which the message originated. As it is not always applicable the default is *N/A*.

os_version

z/OS system product level of the z/OS system on which the message was generated.

Table 10. Mapping SA z/OS Messages to SA z/OS Events

SA z/OS Message	Tivoli Event Class	Tivoli Severity
AOF200I	SA390MSG_CmdUnexpectedCmdResult	CRITICAL
AOF201I	SA390MSG_ClistFailed	CRITICAL
AOF203I	SA390MSG_ClistAuthorizationFailed	WARNING
AOF204I	SA390MSG_ClistInvalidMissingParms	WARNING
AOF205A	SA390MSG_ClistFailedTimerExpired	CRITICAL
AOF206I	SA390MSG_CmdFailed	CRITICAL
AOF211E	SA390MSG_ACFNoAutoOps	CRITICAL

SA z/OS Messages, Alerts and Tivoli Events

Table 10. Mapping SA z/OS Messages to SA z/OS Events (continued)

SA z/OS Message	Tivoli Event Class	Tivoli Severity
AOF214A	SA390MSG_EnvironmentNotInitialized	CRITICAL
AOF216I	SA390MSG_EnvironmentNotInitialized	CRITICAL
AOF218I	SA390MSG_CmdFailedACFSaveInProgress	WARNING
AOF221I	SA390MSG_ACFNoPrimaryFocalPoint	WARNING
AOF222I	SA390MSG_ACFNoBackupFocalPoint	WARNING
AOF228I	SA390MSG_NoInactiveResourceInScope	WARNING
AOF230I	SA390MSG_ACFNoTimeoutValue	WARNING
AOF231I	SA390MSG_ACFTimeoutValueInvalid	CRITICAL
AOF232I ¹	SA390MSG_ACFNoTimeoutValue	WARNING
AOF241I	SA390MSG_ClistInvalidMissingParms	CRITICAL
AOF244I	SA390MSG_ShutdownDelayed	HARMLESS
AOF248I	SA390MSG_ShutdownAlreadyInProgress	WARNING
AOF253I	SA390MSG_TimerNotFound	WARNING
AOF255I	SA390MSG_ACFInvalidUserExit	CRITICAL
AOF256I	SA390MSG_ACFNoOptionalExits	CRITICAL
AOF292I	SA390MSG_CannotAssignReplyId	CRITICAL
AOF293I	SA390MSG_NoOutstandingReply	CRITICAL
AOF294I	SA390MSG_SpoolShortageCmdsExhausted	CRITICAL
AOF303I	SA390MSG_RequestedStateAlreadySet	WARNING
AOF305I	SA390MSG_ProcessorNotInControl	WARNING
AOF309I	SA390MSG_TimerForCatchupExecuted	HARMLESS
AOF312E	SA390MSG_AutoFlagAlreadyOff	WARNING
AOF314I	SA390MSG_TimerForCatchupNotExecuted	CRITICAL
AOF315I	SA390MSG_TimerDeleted	HARMLESS
AOF324A	SA390MSG_ACFInvalidEntry	CRITICAL
AOF350E	SA390MSG_ClistFailed	CRITICAL
AOF355E	SA390MSG_ClistFailed	CRITICAL
AOF356E	SA390MSG_ClistFailed	CRITICAL
AOF501I	SA390MSG_ThresholdActionTaken	WARNING
AOF561A	SA390MSG_ACFNoDrainCmds	WARNING
AOF562I	SA390MSG_JES2DrainFailed	CRITICAL
AOF563A	SA390MSG_UnknownResourceType	WARNING
AOF564A	SA390MSG_ShutdownHaltedActiveJobDev	WARNING
AOF565A	SA390MSG_DrainCmdOff	WARNING
AOF568I	SA390MSG_OutBoundGTWStatus	WARNING
AOF569I	SA390MSG_InBoundGTWStatus	WARNING
AOF575A	SA390MSG_RecoveryNotInProgress	CRITICAL
AOF576A	SA390MSG_RecoveryAutoFlagOff	WARNING
AOF577E	SA390MSG_ThresholdRecoveryHalted	WARNING
AOF580I	SA390MSG_AutoFlagOff	CRITICAL

SA z/OS Messages, Alerts and Tivoli Events

Table 10. Mapping SA z/OS Messages to SA z/OS Events (continued)

SA z/OS Message	Tivoli Event Class	Tivoli Severity
AOF582A	SA390MSG_RecoveryFailed	CRITICAL
AOF583I	SA390MSG_AutoFlagOff	CRITICAL
AOF584I	SA390MSG_ACFNoCmds	CRITICAL
AOF585I	SA390MSG_AutomationAlreadyInProgress	WARNING
AOF587I	SA390MSG_ThresholdRecoveryContinued	WARNING
AOF591E	SA390MSG_RecoveryAutoFlagOff	WARNING
AOF593A	SA390MSG_ShutdownCmdsExhausted	CRITICAL
AOF608A	SA390MSG_WarmStartFailure	CRITICAL
AOF609A	SA390MSG_InvalidReply	MINOR
AOF701A	SA390MSG_MsgTableNotLoaded	CRITICAL
AOF703E ¹	SA390MSG_CmdsExhausted	CRITICAL
AOF719I	SA390MSG_ClistUnexpectedCmdResult	CRITICAL
AOF739I	SA390MSG_ACFNoJESSSubsystem	CRITICAL
AOF745E	SA390MSG_ShutdownNoFinalMsg	WARNING
AOF746A	SA390MSG_ShutdownSuspended	CRITICAL
AOF747A	SA390MSG_ShutdownStuck	CRITICAL
AOF748E	SA390MSG_ShutdownNoFinalMsgResume	WARNING
AOF750I	SA390MSG_ShutdownCmdRplyFailed	CRITICAL
AOF752E	SA390MSG_ShutdownAutoFlagOff	CRITICAL
AOF759A	SA390MSG_RecoveryACFCMDFailed	CRITICAL
AOF760E	SA390MSG_ClistTrapped	CRITICAL
AOF766I	SA390MSG_ACFEntryMissed	CRITICAL
AOF768I	SA390MSG_ACFEntryMissed	CRITICAL
AOF793E	SA390MSG_ACFInconsistentDependencies	CRITICAL
AOF797A	SA390MSG_ACFLoadStopped	CRITICAL
AOF859I	SA390MSG_ACFInstructionIncomplete	CRITICAL
AOF860I	SA390MSG_ACFInstructionRejected	WARNING
AOF861I	SA390MSG_ACFRetractRejected	WARNING
AOF862I	SA390MSG_ACFBadInstruction	CRITICAL
AOF863I	SA390MSG_ACFDuplicateAssert	WARNING
AOF864I ¹	SA390MSG_ACFModifyToAssert	WARNING
AOF865I ¹	SA390MSG_ACFInheritedDataModified	WARNING
AOF866I	SA390MSG_ACFInvalidKeyword	WARNING
AOF867I	SA390MSG_ACFInconsistentData	CRITICAL
AOF868I	SA390MSG_ACFNoJESSSubsystem	CRITICAL
AOF905I	SA390MSG_XCFCommandFailed	WARNING
AOF906I	SA390MSG_MVSCCommandFailed	WARNING
AOF914E	SA390MSG_RecoveryFailed	CRITICAL
AOF924A	SA390MSG_RecoveryFailed	CRITICAL
AOF927E	SA390MSG_RecoveryFailed	CRITICAL

Table 11. SA z/OS Events and Corresponding Mapped SA z/OS Messages (continued)

Tivoli Event Class	SA z/OS Message	Tivoli Severity
SA390MSG_ACFInstructionIncomplete	AOF859I	CRITICAL
SA390MSG_ACFInvalidEntry	AOF324A	CRITICAL
SA390MSG_ACFInvalidUserExit	AOF255I	CRITICAL
SA390MSG_ACFNoAutoOps	AOF211E	CRITICAL
SA390MSG_ACFNoBackupFocalPoint	AOF222I	WARNING
SA390MSG_ACFNoCmds	AOF584I	CRITICAL
SA390MSG_ACFNoDrainCmds	AOF561A	WARNING
SA390MSG_ACFNoJESSubsystem	AOF739I AOF868I	CRITICAL
SA390MSG_ACFNoOptionalExits	AOF256I	CRITICAL
SA390MSG_ACFNoPrimaryFocalPoint	AOF221I	WARNING
SA390MSG_ACFNoTimeoutValue	AOF230I AOF232I	WARNING
SA390MSG_ACFTimeoutValueInvalid	AOF231I	CRITICAL
SA390MSG_ACFUpdateInvalid	base class	
SA390MSG_ACFBadInstruction	AOF862I	CRITICAL
SA390MSG_ACFDuplicateAssert	AOF863I	WARNING
SA390MSG_ACFInheritedDataModified	AOF865I	WARNING
SA390MSG_ACFInstructionRejected	AOF860I	WARNING
SA390MSG_ACFInvalidKeyword	AOF866I	WARNING
SA390MSG_ACFModifyToAssert	AOF864I	WARNING
SA390MSG_ACFRetractRejected	AOF861I	WARNING
SA390MSG_AutoFlagAlreadyOff	AOF312E	WARNING
SA390MSG_AutoFlagOff	AOF580I AOF583I	CRITICAL
SA390MSG_AutomationAlreadyInProgress	AOF585I	WARNING
SA390MSG_CannotAssignReplyId	AOF292I	CRITICAL
SA390MSG_Clist	base class	
SA390MSG_ClistAuthorizationFailed	AOF203I	WARNING
SA390MSG_ClistFailed	AOF201I AOF350E AOF355E AOF356E	CRITICAL
SA390MSG_ClistFailedTimerExpired	AOF205A	CRITICAL
SA390MSG_ClistInvalidMissingParms	AOF204I AOF241I	WARNING
SA390MSG_ClistTrapped	AOF760E	CRITICAL
SA390MSG_ClistUnexpectedCmdResult	AOF719I	CRITICAL
SA390MSG_CmdFailed	AOF206I	CRITICAL
SA390MSG_CmdFailedACFSaveInProgress	AOF218I	WARNING
SA390MSG_CmdsExhausted	AOF703E	CRITICAL

SA z/OS Messages, Alerts and Tivoli Events

Table 11. SA z/OS Events and Corresponding Mapped SA z/OS Messages (continued)

Tivoli Event Class	SA z/OS Message	Tivoli Severity
<i>SA390MSG_CmdUnexpectedCmdResult</i>	AOF200I	CRITICAL
<i>SA390MSG_DrainCmdOff</i>	AOF565A	WARNING
<i>SA390MSG_EnvironmentNotInitialized</i>	AOF214A AOF216I	CRITICAL
<i>SA390MSG_InBoundGTWStatus</i>	AOF569I	WARNING
<i>SA390MSG_InvalidReply</i>	AOF609A	MINOR
<i>SA390MSG_JES2DrainFailed</i>	AOF562I	CRITICAL
<i>SA390MSG_MsgTableNotLoaded</i>	AOF701A	CRITICAL
<i>SA390MSG_NoInactiveResourceInScope</i>	AOF228I	WARNING
<i>SA390MSG_NoOutstandingReply</i>	AOF293I	CRITICAL
<i>SA390MSG_OutBoundGTWStatus</i>	AOF568I	WARNING
<i>SA390MSG_ProcessorNotInControl</i>	AOF305I	WARNING
<i>SA390MSG_RequestedStateAlreadySet</i>	AOF303I	WARNING
<i>SA390MSG_SpoolShortageCmdsExhausted</i>	AOF294I	CRITICAL
<i>SA390MSG_SubsystemRecovery</i>	base class	
SA390MSG_RecoveryACFCMDFailed	AOF759A	CRITICAL
SA390MSG_RecoveryAutoFlagOff	AOF576A AOF591E	WARNING
SA390MSG_RecoveryFailed	AOF582A	CRITICAL
SA390MSG_RecoveryNotInProgress	AOF575A	CRITICAL
<i>SA390MSG_SubsystemShutdown</i>	base class	
SA390MSG_ShutdownAlreadyInProgress	AOF248I	WARNING
SA390MSG_ShutdownAutoFlagOff	AOF752E	CRITICAL
SA390MSG_ShutdownCmdsExhausted	AOF593A	CRITICAL
SA390MSG_ShutdownCmdRplyFailed	AOF750I	CRITICAL
SA390MSG_ShutdownDelayed	AOF244I	HARMLESS
SA390MSG_ShutdownHaltedActiveJobDev	AOF564A	WARNING
SA390MSG_ShutdownNoFinalMsg	AOF745E	WARNING
SA390MSG_ShutdownNoFinalMsgResume	AOF748E	WARNING
SA390MSG_ShutdownStuck	AOF747A	CRITICAL
SA390MSG_ShutdownSuspended	AOF746A	CRITICAL
<i>SA390MSG_Threshold</i>	base class	
SA390MSG_ThresholdActionTaken	AOF501I	WARNING
SA390MSG_ThresholdRecoveryContinued	AOF587I	WARNING
SA390MSG_ThresholdRecoveryHalted	AOF577E	WARNING
<i>SA390MSG_Timer</i>	base class	
SA390MSG_TimerDeleted	AOF315I	HARMLESS
SA390MSG_TimerForCatchupExecuted	AOF309I	HARMLESS
SA390MSG_TimerForCatchupNotExecuted	AOF314I	CRITICAL
SA390MSG_TimerNotFound	AOF253I	WARNING

Table 11. SA z/OS Events and Corresponding Mapped SA z/OS Messages (continued)

Tivoli Event Class	SA z/OS Message	Tivoli Severity
SA390MSG_UnknownResourceType	AOF563A	WARNING
SA390MSG_WarmStartFailure	AOF608A	CRITICAL

Mapping NetView Alerts to Tivoli Events

SA z/OS processor operations indicates severe failures which require operator intervention by issuing NetView alerts.

These alerts are forwarded to the Tivoli Enterprise Console exploiting the alert adapter. The generic mapping of the NetView generic alerts applies, that is, no new event classes are introduced, but those defined by the NetView z/OS alert adapter.

The only attribute default changed in case of alerts issued by SA z/OS processor operations is the default of the attribute *sub_source*. While the adapter fills in *NET* as default, the *sub_source* attribute has the value *SAOS390_ProcOps* in case of alerts issued by SA z/OS.

The following table describes the resulting mapping of SA z/OS processor operations alerts to the Tivoli Enterprise Console event classes:

Table 12. Mapping SA z/OS Alerts to SA z/OS Events

AlertId	SA z/OS Processor Operations Alert	Tivoli Event Class
1	COMMUNICATION TASK FAILED	SNA_Software_Program_Error
2	LOST PS/2 [®] SESSION	SNA_Link_Error
3	DCCF DETECTED	SNA_Software_Program_Error
4	WAIT STATE DETECTED	SNA_Software_Program_Error
5	CONSOLE LOST	SNA_Input_Output_Device_Error
6	CONSOLE ACCESS LOST	SNA_Input_Output_Device_Error
7	COULD NOT OBTAIN SYSTEM CONSOLE	SNA_Input_Output_Device_Error
8	TARGET HARDWARE PROBLEM	SNA_Equipment_Malfunction

TEC Rules for SA z/OS Events

A selected set of SA z/OS messages is mapped to Tivoli events and forwarded to the Tivoli Enterprise Console. Those events can be filtered at the Tivoli Enterprise Console, correlated with other events, and/or trigger an action.

With TEC Notification by SA z/OS you receive a set of sample rules correlating SA z/OS events.

The following sections list the SA z/OS messages for which correlation rules are provided and the semantics of those rules is shortly described.

Correlation Rules for SA z/OS Message AOF564A

SA z/OS Message

```
AOF564A time : { JES2 SHUTTING DOWN - txt - OPERATOR INTERVENTION
REQUIRED | title | job_names | task_names | device_names }
```

SA z/OS Messages, Alerts and Tivoli Events

Where *'txt'* is one of:

```
| DEVICES NOT DRAINED  
| JOBS STILL EXECUTING  
| TASKS STILL EXECUTING  
| JOBS STILL EXECUTING, DEVICES NOT DRAINED  
| TASKS STILL EXECUTING, DEVICES NOT DRAINED  
| JOBS AND TASKS STILL EXECUTING  
| JOBS AND TASKS STILL EXECUTING TO DEVICES NOT DRAINED
```

and *'title'* is one of:

```
Tasks ...  
Devices ...  
Jobs ...
```

Description

JES2 is being shutdown and there are jobs, tasks and/or devices that have not been drained. This is indicated by :

```
AOF564A time : JES2 SHUTTING DOWN - txt -  
OPERATOR INTERVENTION REQUIRED.
```

This message is followed by a number of messages of type

```
| AOF564A time : title  
| AOF564A time : {device_names | job_names | task_names }
```

detailing the jobs or tasks or devices causing the shutdown halt.

Event Class

SA390MSG_ShutdownHaltedActiveJobDev

Event Severity

WARNING

Correlation Rules:

```
sa390msg_related_564_rule1  
sa390msg_related_564_rule2  
sa390msg_related_564_rule3  
sa390msg_related_564_rule4  
sa390msg_related_564_rule5  
sa390msg_related_564_rule6
```

These rules link events originating from messages:

```
| AOF564A time : {device_names | job_names | task_names }
```

to the event originating from a message

```
AOF564A time : JES2 SHUTTING DOWN - txt - OPERATOR INTERVENTION REQUIRED.
```

Further devices or jobs or tasks listed in the above message are copied to attribute *user2* of the event originating from

```
AOF564A time : JES2 SHUTTING DOWN - txt - OPERATOR INTERVENTION REQUIRED.
```

When events originating from messages of type

```
AOF564A time : title
```

are received, the information on the type of resource causing the shutdown halt is added to attribute *user1* of the event originating from

```
AOF564A time : JES2 SHUTTING DOWN - txt - OPERATOR INTERVENTION REQUIRED.
```

If the status of an event originating from a message of type

AOF564A time : {device_names | job_names | task_names }

is changed, this is propagated to the linked message originating from a message of type

AOF564A time : JES2 SHUTTING DOWN - txt - OPERATOR INTERVENTION REQUIRED.

by adding this information into attributes *user3* and *user4*.

Operation of TEC Notification by SA z/OS

This chapter describes the use of TEC Notification by SA z/OS. It is separated in the following sections describing the appropriate aspects of using TEC Notification by SA z/OS:

- “Startup of TEC Notification by SA z/OS”
- “How TEC Notification by SA z/OS Works” on page 253
- “Termination of TEC Notification by SA z/OS” on page 256

The starting point for all Tivoli-related operations after a successful startup of TEC Notification by SA z/OS is the Tivoli desktop.

Startup of TEC Notification by SA z/OS

This section lists:

- the actions you need to perform to prepare the monitored z/OS systems for routing messages and alerts to Tivoli (see “Host Part”)
- the actions that need to be performed on the monitoring Tivoli Enterprise Console to let you see incoming events displayed (see “Tivoli Enterprise Console Part” on page 252).

Host Part

Prerequisites:

- Tivoli NetView for z/OS is up and running.
- Automation table with fragment *INGMTEC* included is active.
- System Automation for z/OS is initialized.

If all prerequisites are fulfilled, the autotask has been started and the notification classes for the autotask have been set. Message *AOF646I (Tivoli Enterprise Console Notification Service is initialized)* is issued to indicate that the system is correctly customized and initialized.

Actions and System Responses:

1. On the z/OS focal point system where the message adapter is running, you need to start the Event/Automation Service. Issue the command:

```
S IHSAEVNT
```

If you issue this command as shown here without any options, the Event/Automation Service will use the defined initialization and adapter configuration files. For more information on this command, see *Tivoli NetView for z/OS Customization Guide*.

2. As a result of the previous step, the message adapter and, if configured, the alert adapter become initialized. When these are completed, this is indicated by the *IHS0124I* messages:

```
IHS0124I: Message Adapter task initialization complete.
```

```
IHS0124I: Alert Adapter task initialization complete.
```

Operation of TEC Notification by SA z/OS

3. Message *IHS0124I* for the message adapter triggers the execution of the Notification Service initialization procedure *INGRTIN*. This procedure sets the environment global variables and as a result of this procedure, the routing of SA z/OS messages to TEC is enabled. This is indicated by message *AOF642I*:
TIVOLI ENTERPRISE CONSOLE NOTIFICATION SERVICE IS ENABLED.
4. The alert routing must be enabled manually. Either on the SA z/OS processor operations focal point system or on the SA z/OS system operations focal point system, depending on where the alert adapter is running, you need to issue the NetView command

```
NPDA SRFILTER TECROUTE PASS E NTFY
```

to set the TECROUTE filter for alerts of type *NOTIFICATION* to **PASS**.

The message

```
BNJ1341I SRF/SRFILTER COMMAND ACCEPTED
```

is returned. The correct setting for the filter can be verified by issuing the command *NPDA DFILTER TECROUTE*. The output shown below should be displayed on the NetView console with the action for the event type *NTFY* set to **PASS**.

```
N E T V I E W          SESSION DOMAIN: IPUFJ   BALI   06/10/97 10:33:57
NPDA-20A              * CURRENT FILTER STATUS *          REC 1 TO 2 OF 2
                      FILTER TYPE: ALERT FORWARDING TO T/EC
```

```
SEL# ACTION DATA  ETYPE FTYPE  ----- RESNAME, TYPE, OR ADAPTADR -----
( 1) PASS   ..... NTFY
**** DEFAULT ACTION **** BLOCK ****
```

Tivoli Enterprise Console Part

Prerequisites:

1. Enter the appropriate Tivoli command to have the Tivoli desktop up and running.
2. TEC event server is started. If this is not the case, you can start it, for example, from the *Start-up* pull-down menu of the *EventServer* icon.

Sequence:

1. Start the Tivoli Enterprise Console. Now your Tivoli desktop could look similar to Figure 138 on page 253.

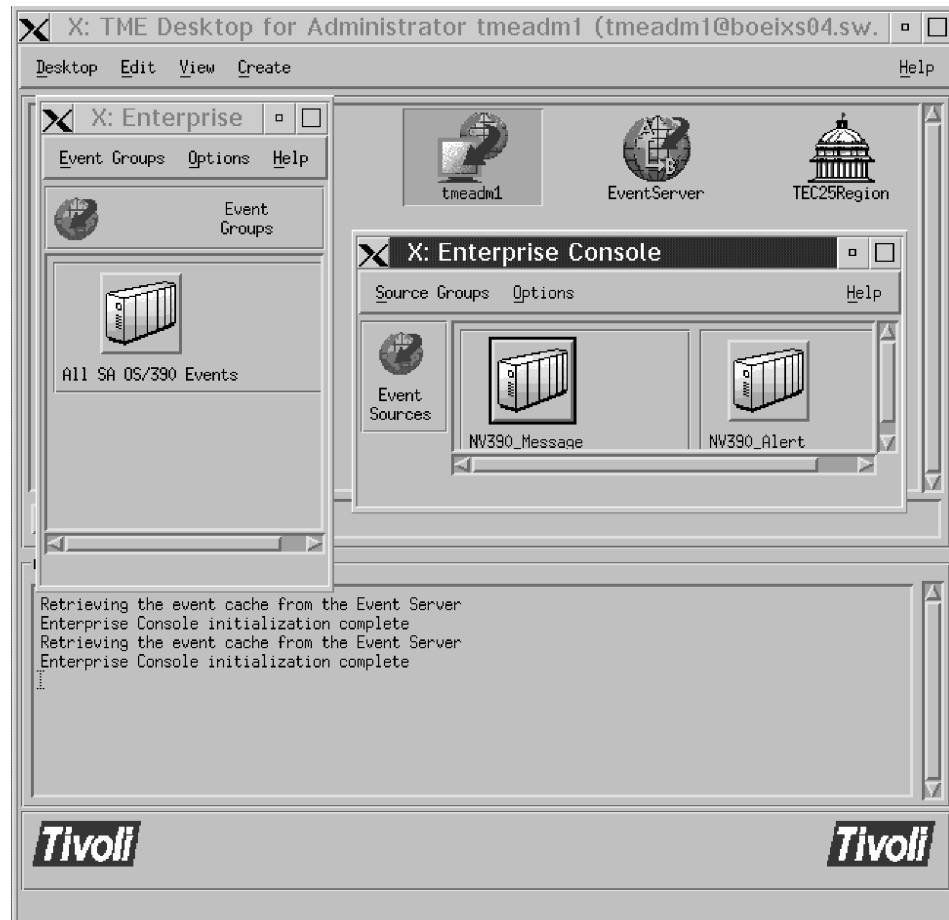


Figure 138. Tivoli desktop with started Tivoli Enterprise Console

2. Open the event groups specific for SA z/OS to have incoming events displayed by double-clicking on the appropriate icon. In our sample Tivoli Enterprise Console from Figure 138, you would open the event group **All SA OS/390 Events**.

How TEC Notification by SA z/OS Works

Host Part

SA z/OS issues messages using the module *AOFMSG* which generates and forwards the messages to the registered auto operator task *%AOFTECTASK%*. This autotask processes the automation table fragment *INGMTEC*. If the message matches the filter criteria, it is either directly provided to the local message adapter (in case you work in a local configuration) or it is sent to the focal point system (for a distributed configuration). You find a description of both the local and the distributed environment in *System Automation for z/OS Planning and Installation*.

SA z/OS alerts are forwarded to the Networking NetView focal point system and provided to the alert adapter, which routes them to TEC.

Tivoli Enterprise Console Part

Incoming events are displayed in the corresponding event groups (**All SA OS/390 Events**). Predefined correlation rules are applied if applicable. TEC Notification by SA z/OS, provides a Tivoli task that enables you to invoke the NetView 3270 Management Console from the Tivoli Enterprise Console - no matter where the

Operation of TEC Notification by SA z/OS

NetView 3270 Management Console is installed. As shown in Figure 139 you select an event for which you want to invoke the NetView 3270 Management Console and from the **Task** pull-down choice, you select action **Execute on Selected Event...**

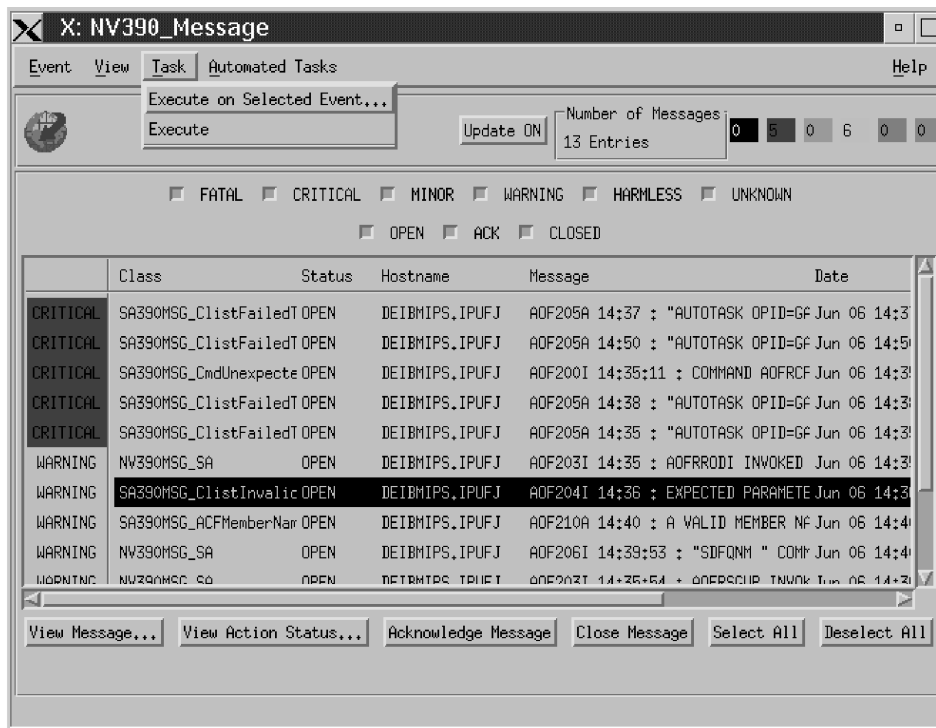


Figure 139. Invoking Predefined Actions for Events

This brings up another dialog, called the *TEC Tasks* dialog, shown in Figure 140 on page 255. From this dialog, you can now invoke the NetView 3270 Management Console:

- Double click on a task library (in our example: *SAOS390 Tasks*).
- Double click on a task (in our example: *Start NetView Console*).
- Optionally you can change the default configuration parameters for the startup of the NetView 3270 Management Console.
- Set the push button beneath the list of configuration parameters to *Execute Local*.
- Press the *Execute and Close* push button.

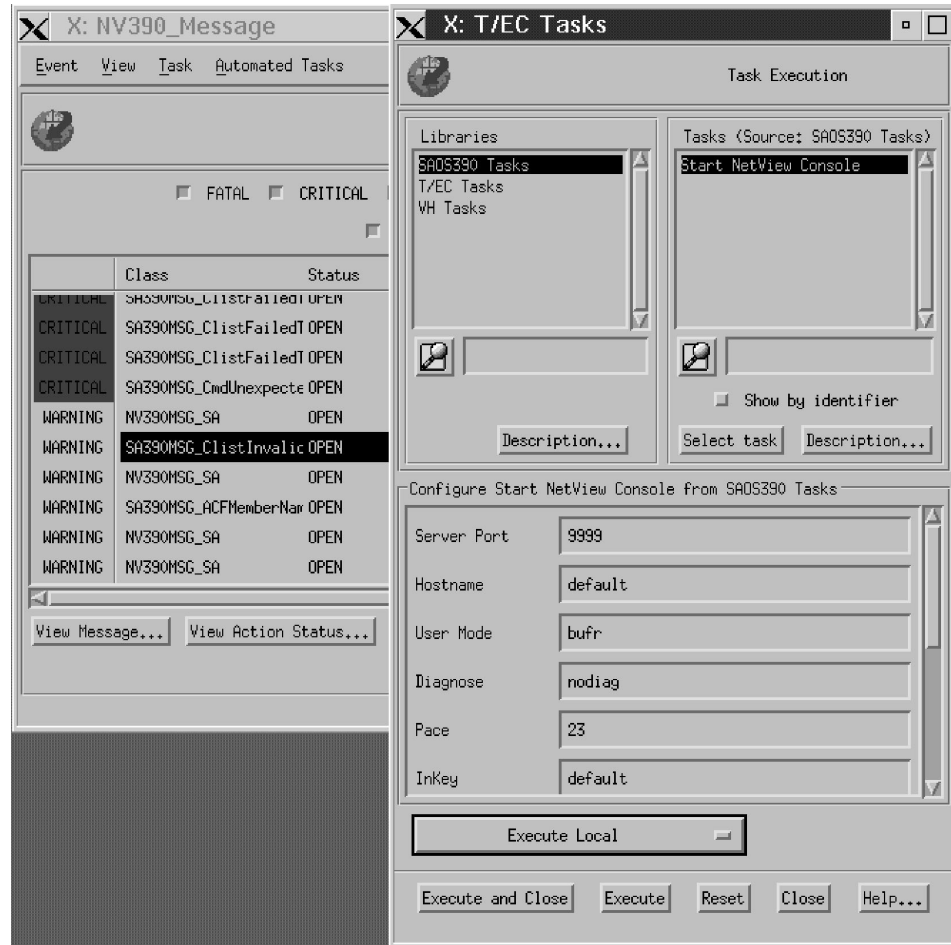


Figure 140. TEC Tasks Dialog

An example of a NetView 3270 Management Console screen is shown in Figure 141 on page 256. Before you reach this screen, a *NetView Logon Panel* prompts you to enter your operator ID and the password.

With the NetView 3270 Management Console you can logon to the NetView of the z/OS system which, as the event source, triggered the event. In this way, you as a system operator or Tivoli administrator can repair damages on the monitored z/OS system directly from your Tivoli Enterprise Console by entering NetView commands into the NetView 3270 Management Console.

Operation of TEC Notification by SA z/OS

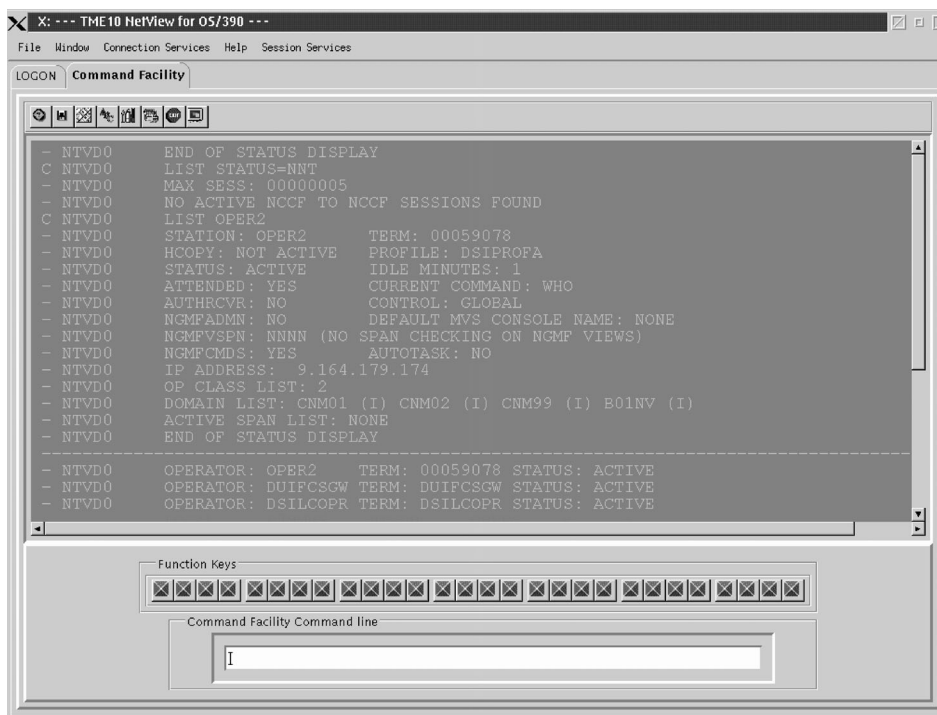


Figure 141. NetView 3270 Management Console

Termination of TEC Notification by SA z/OS

Host Part

Terminating the SA z/OS message notification to TEC is achieved by deregistering the auto operator task *AUTOTEC* as a notification target for SA z/OS messages. You cancel this notification for the auto operator *AUTOTEC* using the NetView command:

```
| /AUTOTEC: INGNTFY OFF
```

Terminating SA z/OS alerts notification to TEC is achieved by setting the *TECROUTE* filter to *BLOCKED*:

```
| NPDA SRFILTER TECROUTE BLOCK E NTFY
```

Tivoli Enterprise Console Part

After terminating the notification of SA z/OS messages and NetView alerts on the host, you will only see incoming traditional Tivoli events on the Tivoli Enterprise Console.

Part 5. Appendixes

Appendix A. Automation Initialization Options

At the start of SA z/OS initialization a multiline write-to-operator (MLWTO) message, AOF767I, is issued describing the options available and system default names and actions. The lines of the MLWTO define the options that can be specified for SA z/OS initialization. Each line is listed and explained below.

AOF767I is followed by either an AOF603D or an AOF606D WTOR. SA z/OS replies automatically to an AOF603D after two minutes unless this has been changed as a part of your site's automation policy. SA z/OS does not automatically reply to an AOF606D message.

The appearance of AOF767I, AOF603D and AOF606D can be suppressed by setting the variable AOFINITREPLY to 0 (see appendix *Global Variables to Enable Advanced Automation in System Automation for z/OS Defining Automation Policy*).

A description of the options follows:

. STOP - CANCEL AUTOMATION

This line is always present. It indicates that you can reply **STOP**. Doing so causes SA z/OS to issue an AOF605I and stop running. You must recycle NetView to restart SA z/OS.

. PAUSE - SUSPEND AUTOMATION

This line is present unless SA z/OS initialization has previously been paused. It indicates that you can reply **PAUSE**. Doing so causes SA z/OS to go into a paused state. An AOF604I message is issued, the AOF767I message is reissued, and an AOF606D WTOR is issued. You should not automate the reply to the AOF606D WTOR, as this disables the pause function.

If the automation manager is (re-)started or refreshed (via the INGAMS or MVS modify command) while the automation agent is paused, the reply is cancelled and the automation agent initialization is redriven from the beginning.

. TABLE= - OVERRIDE DEFAULT AUTOMATION TABLE

This line is always present. It indicates that you can reply **TABLE=new_table**. Doing so causes SA z/OS to load the NetView automation table *new_table* rather than the ones whose names are specified in the automation control file being loaded. The NetView automation table name specified here overrides a table name stored in the warm start cache. The NetView automation table name specified must exist. If SA z/OS is unable to load the automation table it issues an AOF701I message telling you this, and an AOF715D WTOR asking you to specify the name of an alternative automation table.

. NOSTART - DO NOT AUTOMATE SUBSYSTEM STARTUP

This line is always present. It indicates that you can reply **NOSTART**. Doing so causes SA z/OS to initialize the status of all subsystems to AUTODOWN, rather than DOWN and set the resource's hold flag on the automation manager to YES. This means that after initial status determination is complete, SA z/OS does not go on to start the subsystems. When you do want the subsystems started, issue an appropriate SA z/OS command to achieve this. If VTAM is up you can log onto NetView and issue the commands there.

. NOSAVE - DO NOT SAVE DATA TO CACHE

This line is present if the DSISVRT task, which is used for the warm start

Automation Initialization Options

cache, is operational. It indicates that you can reply **NOSAVE**. Replying **NOSAVE** instructs SA z/OS not to save the data loaded from the automation control file into the warm start cache. The default action (if you do not reply **NOSAVE**) after a cold start is to save the data into the warm start cache.

. **WARM - PERFORM A WARM START FROM CACHE**

This line is present if SA z/OS has determined that the warm start cache is available and contains data from an automation control file that was generated by the SA z/OS customization dialog. It indicates that you can reply **WARM**. Doing so causes SA z/OS to start from the warm start cache.

. **REFRESH - PERFORM A WARM START FROM CACHE, REFRESH CHANGED DATA**

This line is present if SA z/OS has determined that the warm start cache is available and contains data from an automation control file that was generated by the SA z/OS customization dialog. This indicates that SA z/OS has detected changed data on disk. Specifying **REFRESH** means that SA z/OS starts from the warm start cache and picks up changed data from the disk.

. **COLD - PERFORM A COLD START FROM DISK**

If you specify **COLD**, SA z/OS will start with data from the disk and will not use any data from the cache.

. **ENTER - CONTINUE WITH *default***

where *default* is one of **WARM**, **REFRESH** or **COLD**. Refer to the appropriate description in this list.

NOTE: COLD START NOT POSSIBLE, NO ACF MEMBER AVAILABLE

This line is present if no valid automation control file has been found on disk. If the warm start cache is valid (that is, the cache timestamp token matches the Automation Manager token) then a **WARM** start will be attempted.

NOTE: WARM START NOT POSSIBLE, DSISVRT NOT AVAILABLE

This line is present if SA z/OS has determined that the warm start cache is unavailable. It indicates that the DSISVRT task could not be started.

NOTE: NO USABLE ACF IN CACHE

This line is present if SA z/OS has determined that the warm start cache is available, but the data in it cannot be used.

NOTE: CACHE NEEDS TO BE REFRESHED

This line is present if SA z/OS has determined that the warm start cache is available, but needs refreshing. If the ACF updates will only affect the Automation Agent then a **COLD** start will be the default.

If the ACF updates affect the Automation Manager then a **WARM** start will be the default. When ACF updates affect the Automation Manager an **INGAMS REFRESH** is required before a **COLD** start is possible.

NOTE: REFRESH REQUIRED BUT NOT POSSIBLE

This line is present if SA z/OS has determined that the warm start cache is available, and needs refreshing, but the ACF member is invalid or missing. In this case SA z/OS will proceed with a **WARM** start from cache.

NOTE: CACHE IS CURRENT, NO UPDATE REQUIRED

This line is present if SA z/OS has determined that the warm start cache is available, and does not need refreshing.

NOTE: DEFAULT IS: *default*

This line indicates the default (one of **WARM**, **REFRESH** or **COLD**).

A sample of how the AOF767I message could look, is in Figure 142.

```
AOF767I AUTOMATION OPTIONS: 497
. STOP      - CANCEL AUTOMATION
. PAUSE     - SUSPEND AUTOMATION
. NOSTART   - DO NOT AUTOMATE SUBSYSTEM STARTUP
. NOSAVE    - DO NOT SAVE AUTOMATION CONTROL FILE
. COLD      - COLD START FROM DISK
. TABLE=   - OVERRIDE DEFAULT AUTOMATION TABLE(S)
. ENTER     - CONTINUE WITH 'COLD'
NOTE: NO USABLE ACF IN CACHE
NOTE: DEFAULT IS: COLD
68 AOF603D ENTER AUTOMATION OPTIONS - DOMAIN IPSNO
```

Figure 142. Sample AOF767I Message

If the message on your screen looks significantly different from the sample shown, ask your automation programmer whether he used the initialization exit *AOFEXI01* to modify the message.

Appendix B. Using the I/O Operations ISPF Dialog

Notes:

1. Refer to *System Automation for z/OS Planning and Installation* for instructions on how to install and to modify the ISPF dialog.
2. The IBM-supplied REXX code, including the EXECs, panels, messages, and tables, is supported, provided it has not been modified. Therefore, if you intend to modify IBM-supplied code, it is advisable to retain two copies: the IBM-supplied copy and the modified
3. Refer to the ISPF publications listed in Table 2 on page xiv for more information on ISPF.

Getting Started

To use the I/O operations ISPF dialog:

- You must already be logged on to your user ID and using ISPF.
- The I/O operations ISPF dialog must be available.

To see whether the I/O operations ISPF dialog is available, display the ISPF/PDF *Primary Option* panel on the ISPF main menu.

- If I/O operations is displayed as an option on the ISPF main menu, select it.

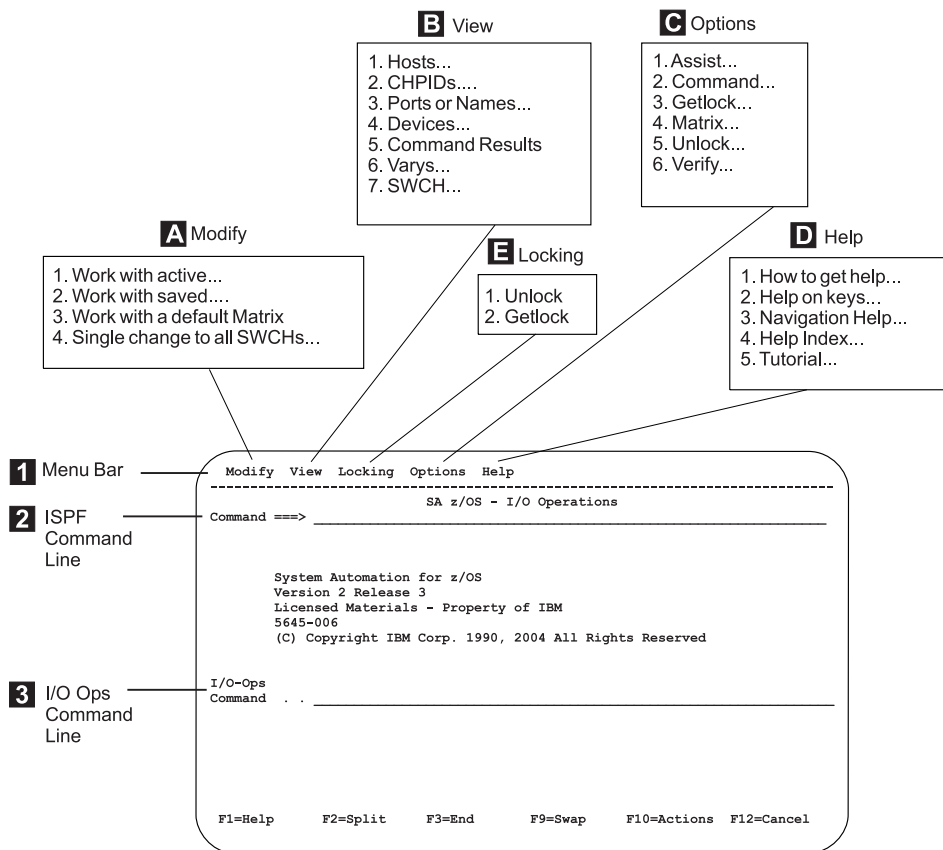
The I/O operations ISPF main menu, which is shown on the next page, should now be displayed. If it is not, consult your system administrator or *System Automation for z/OS Planning and Installation* for further instructions.

- If I/O operations is not displayed on the ISPF/PDF *Primary Option* panel:
 1. Select option 6 (Command) on the ISPF/PDF *Primary Option* panel and press **Enter**.
 2. If the library required by the customization dialog (ING.CUSTOM.IHVCONF) has been allocated using the TSO logon procedure or using a CLIST of your own, type **INGDLG SELECT (IOCONNECT) ALLOCATE (NO)**

(To issue the INGDLG command, the data set ING.SINGIREX must be allocated in the SYSPROC DD statement of your logon procedure.)

The I/O operations ISPF main menu, which is shown on the following page, should now be displayed. If it is not, or if your TSO logon procedure is not set up to allocate the data set required by the ISPF dialog, consult your system administrator or *System Automation for z/OS Planning and Installation* for further instructions.

Starting with the I/O Operations ISPF Main Menu



1 Menu Bar from which you select menus **A** through **E**

A Modify Menu

lets you modify a switch configuration in matrix format and issue a change to all switches allocated to the issuing SA z/OS I/O operations. Place the cursor underneath **Modify** and press **Enter**. See "A. If You Select the Modify Menu" on page 265 for more information.

B View Menu

prompts you for the operands of the Display commands and then displays the data you request. Place the cursor underneath **View** and press **Enter**. See "B. If You Select the View Menu" on page 267 for more information.

C Options Menu

lets you customize the IBM-supplied defaults for your user ID. Place the cursor underneath **Options** and press **Enter**. See "C. If You Select the Options Menu" on page 270 for more information.

D Help Menu

shows you how to display the online help and the online tutorial. Place the cursor underneath **Help** and press **Enter**. See "D. If You Select the Help Menu" on page 271 for more information.

E Locking Menu

is a simple way to enter the Getlock and Unlock commands. Enter **1** or **2**. If access is controlled by RACF, the UNLOCK command requires only the Read authorization, but the GETLOCK command requires the Control

level. See *System Automation for z/OS Operator's Commands* for information about authorization required for I/O operations commands and for information about the GETLOCK and UNLOCK commands.

2 ISPF Command Line

On the ISPF command line, you can enter any applicable ISPF command. For more information on ISPF, see the ISPF publications listed in Table 2 on page xiv.

3 I/O Operations Command Line

On the I/O operations command line, you can enter any SA z/OS I/O operations command. See *System Automation for z/OS Operator's Commands* for a complete list of these commands.

A. If You Select the Modify Menu

View

- | |
|--|
| <ol style="list-style-type: none"> 1. Hosts... 2. CHPIDs.... 3. Ports or Names... 4. Devices... 5. Command Results 6. Varys... 7. SWCH... |
|--|

Select A Subtask	Action (Press Enter When Needed)
1. Work with active (matrix)...	<ul style="list-style-type: none"> • a. Select a switch from the list on the <i>Select SWCH</i> panel. • b. Go to "A Sample Way to Modify a Switch Matrix" on page 266 for more information.
2. Work with saved (matrix)...	<ul style="list-style-type: none"> • a. Select a saved configuration from either: <ul style="list-style-type: none"> – Host (saved in ISPF) – Switch • b. Follow directions.
3. Work with a default Matrix	<ul style="list-style-type: none"> • Go to "A Sample Way to Modify a Switch Matrix" on page 266 for more information.
4. Single change to all SWCHs...	<ul style="list-style-type: none"> • a. Select a command on the <i>Single Change to All SWCHs</i> panel. • b. Enter the required information in the fields on the panel (for example, enter a port address or name for the Block command). • c. Press F6. • d. If the <i>Verify</i> panel is displayed, verify the information and press Enter. • e. When processing is complete, do one of the following: <ul style="list-style-type: none"> – Repeat this sequence of steps to make another change. – Press F3 to exit.

After you have created or changed a switch configuration in matrix format, you can activate it or make it operational on the designated switch, or you may save it on the host or on a selected ESCON Director. The issuing SA z/OS I/O operations must be running and the switch must be allocated to it.

A. If You Select the Modify Menu

You cannot activate the IBM-supplied default matrix until you have changed its default switch device number of 0000 to a number that is valid for your configuration. (See the last entry in the table under “Sample Changes You Can Make to a Port.”)

It is very advisable to save an active switch matrix, even if this is not mandatory.

A Sample Way to Modify a Switch Matrix

When a switch matrix is displayed, you can make the following changes on it; you do not have to enter commands on the command line.

1. If you want to assign a name to the switch, enter the name in the *SWCH Name* field.

Note: Press **Enter** if you want to see the change highlighted.

2. To make a change to a port, check whether that port’s horizontal row is displayed on the matrix. If it is not:
 - To scroll down the left column of ports, use **F8** to scroll to the higher port addresses and **F7** to scroll in the other direction (backwards).
 - To scroll across the columns of ports in the matrix, use **F11** to the right across the screen and **PF10** to scroll to the left.

Some sample changes are shown under “Sample Changes You Can Make to a Port.”

3. After you have finished making changes, consider whether you want to save or to activate the switch configuration, or both.

Note: It is strongly recommended that you save a switch configuration before activating it.

4. To save a switch configuration:
 - a. Press **F5**.
 - b. On the *Select Destination of Save Configuration* panel, first choose between saving at a switch or at the host. Then, either select a name from the list that is presented, or enter a new name and description.
 - c. Press **Enter** to save the matrix of the switch configuration.

If you are replacing another switch configuration, another panel will prompt you to verify the change. In that case, press **Enter** again.
 - d. If you want to activate the switch configuration, refer to the next step.
5. To activate the switch configuration:
 - a. Press **F6** on the screen of the matrix.
 - b. Verify the information.
 - c. Press **Enter**.

Sample Changes You Can Make to a Port

I/O operations Command	Action
Block a port	<ul style="list-style-type: none">• Type B at the intersection point between the port row and the B column. For example, if you type B in the B column at D5’s row, D5 will be blocked.
Unblock a blocked port	<ul style="list-style-type: none">• Type U or a space at the intersection point between the port row and the B column. For example, type U over the B in the B column at D5’s row to unblock port D5.

A. If You Select the Modify Menu

I/O operations Command	Action
Connect two ports statically (dedicated)	<ul style="list-style-type: none"> Type the port address of the 2nd port at the intersection point between the 1st port and the C column. For example, type C3 in the C column at port row E4 to connect C3 and E4 statically.
Disconnect a static (dedicated) connection	<ul style="list-style-type: none"> Type spaces over the port address at the intersection point between the port row and the C column. For example, type spaces over C3 in the C column at port row E4 to disconnect C3 and E4 from each other.
Prohibit a dynamic connection	<ul style="list-style-type: none"> Using PF10 and F11 to scroll right and left, display the port column of the 2nd affected port. Type P at the intersection point between the 1st port's row and the 2nd port's column. For example, type P at the intersection of port row D7 and port column E1 to prohibit these ports from connecting dynamically to each other.
Allow a dynamic connection	<ul style="list-style-type: none"> Using PF10 and F11 to scroll right and left, display the port column of the 2nd affected port. Type A at the intersection point between the 1st port's row and the 2nd port's column. For example, type A at the intersection of port row D7 and port column E1 to allow these ports to connect dynamically to each other.
Copy Allows and Prohibits from one port row to another port row	<ul style="list-style-type: none"> Type C on the blank line to the left of the port row you plan to use as a source. Type T on the blank line to the left of the port row you plan to use as a target.
Name or rename a port	<ul style="list-style-type: none"> Type a port name at the intersection point between the port row and the Name column. For example, type PRINTER1 in the Name column of port row C5 to name C5 PRINTER1. It is recommended that the name be in uppercase letters. (Refer to <i>System Automation for z/OS Planning and Installation</i> for suggestions for naming ESCON Director ports.) You can remove a name by typing blanks over it.
Change the switch device number	<ul style="list-style-type: none"> Type a switch device number in the <i>Target SWCH Number</i> field. (This is a required action if you start with the IBM-supplied default matrix of 0000.)

B. If You Select the View Menu

Options

- | |
|--|
| <ol style="list-style-type: none"> 1. Assist... 2. Command... 3. Getlock... 4. Matrix... 5. Unlock... 6. Verify... |
|--|

If you select the *View* menu, the subtasks shown at the left are displayed. When the selected view is displayed:

- Use **F7** to scroll backward and **F8** to scroll forward to view the information.
- Press **F3** to return to the display selection screen.

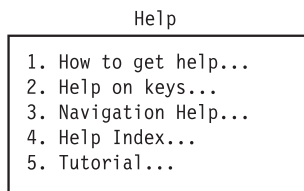
B. If You Select the View Menu

Selection	Option	Action
1. Hosts...	The SA z/OS I/O operations base programs that are known to the issuing SA z/OS I/O operations.	Type * on the 1st blank line. See the DISPLAY HOST command in <i>System Automation for z/OS Operator's Commands</i> .
	Detailed information about the SA z/OS I/O operations base program specified.	Type the VTAM name of an SA z/OS I/O operations on the 1st blank line. See the DISPLAY HOST command in <i>System Automation for z/OS Operator's Commands</i> .
	A list of all SA z/OS I/O operations base programs that have the specified switch allocated to them.	Type a switch device number on the 2nd blank line. See the DISPLAY HOST command in <i>System Automation for z/OS Operator's Commands</i> .
2. CHPIDs...	A list of all CHPIDs with their associated information	Type * on the blank line. See the DISPLAY CHP command in <i>System Automation for z/OS Operator's Commands</i> .
	Detailed information about a specified CHPID	Type a CHPID number on the blank line. See the DISPLAY CHP command in <i>System Automation for z/OS Operator's Commands</i> .
3. Ports...	A list of all ports with the same name	Type the name of a named port on the 1st blank line. A generic port name may be used. See the DISPLAY NAME command in <i>System Automation for z/OS Operator's Commands</i> .
	Details about a port name or port address on a specified switch	Type either a port address on the 2nd blank line or a port name on the 3rd blank line. Then, type a switch device number on the 4th blank line. See the DISPLAY PORT command in <i>System Automation for z/OS Operator's Commands</i> .
	Information about all ports, only blocked ports, or only statically connected ports on a specified switch	Type A , B , or C on the 5th blank line. Then, type a switch device number on the 6th blank line. See the DISPLAY PORT command in <i>System Automation for z/OS Operator's Commands</i> .

B. If You Select the View Menu

4. Devices...	A list of all devices attached to a specified switch	Type * on the 1st blank line. Then, type a switch device number on the 2nd blank line. See the DISPLAY DEVICE command in <i>System Automation for z/OS Operator's Commands</i> .
	Detailed information about a specific device with relation to all switches	Type a device number on the 1st blank line. Then, type * on the 2nd blank line. See the DISPLAY DEVICE command in <i>System Automation for z/OS Operator's Commands</i> .
	Detailed information about a specified device relative to a specified switch	Type a device number on the 1st blank line. Then, type a switch device number on the 2nd blank line. See the DISPLAY DEVICE command in <i>System Automation for z/OS Operator's Commands</i> .
	Multiple devices on a specified switch	Type a range of device numbers on the 1st blank line. Then, type a switch device number on the 2nd blank line. See the DISPLAY DEVICE command in <i>System Automation for z/OS Operator's Commands</i> .
5. Command Results	A list of command results with relation to each SA z/OS I/O operations	See the DISPLAY RESULTS command in <i>System Automation for z/OS Operator's Commands</i> .
6. Varys...	A list of failed vary path requests with relation to a specific SA z/OS I/O operations host	Type the SA z/OS I/O operations VTAM name on the blank line. See the DISPLAY VARY command in <i>System Automation for z/OS Operator's Commands</i> .
7. SWCH...	A list of all switches allocated, or attached, to the issuing instance of I/O operations.	Type * on the blank line. See the DISPLAY SWITCH command in <i>System Automation for z/OS Operator's Commands</i> .
	Details about a specific switch	Type a switch device number on the blank line. See the DISPLAY SWITCH command in <i>System Automation for z/OS Operator's Commands</i> .

C. If You Select the Options Menu



In the I/O operations ISPF dialog, a number of options are shipped with IBM-supplied defaults. You can customize these defaults to your ISPF user ID. Except for the *Verify* option, your customized options remain in effect for your user ID across SA z/OS I/O operations sessions.

1. Do either of the following:
 - On the dialog's main menu, select the *Options*, and select one of the entries.
 - Enter **OPTIONS** in the command area of a dialog panel
2. Make a selection (1—6).
3. If you want to save your change, press **F3** or enter **END**

1. Assist

Describes the return codes and reason codes in the messages. Selections are:

1. To receive explanatory text and specific messages (IBM-supplied default)
2. Not to receive explanatory text on the return codes and reason codes.

2. Command

Specifies a set of default connectivity options. Selections are:

1. NOForce, Vary, BACkout (IBM-supplied defaults)
2. NOForce, Vary, NOBackout
3. NOCheck, Vary, BACkout
4. NOCheck, Vary, NOBackout
5. Force, Vary, NOBackout
6. Force NOVary, NOBackout

3. Getlock

Obtains processing control. (See the GETLOCK command in *System Automation for z/OS Operator's Commands* for more information.) Selections are:

1. Never take control from another user.
2. Be prompted with the *Getlock* panel so you can decide whether you want to take control (IBM-supplied default).
3. Always take control. If a command is being processed by another user, however, processing can be interrupted and the command can fail.

4. Matrix

Specifies which changes on the switch configuration will be activated by SA z/OS I/O operations. (Select them with any character.) Selections are:

- ___ ALL commands (overrides other options; IBM-supplied default)
- ___ Allow and Prohibit
- ___ Connect and Disconnect

C. If You Select the Options Menu

- ___ Block and Unblock
- ___ Port names
- ___ Switch name

5. Unlock

Relinquishes control of SA z/OS I/O operations command processing. (See the UNLOCK command in *System Automation for z/OS Operator's Commands* for further information.) Selections are:

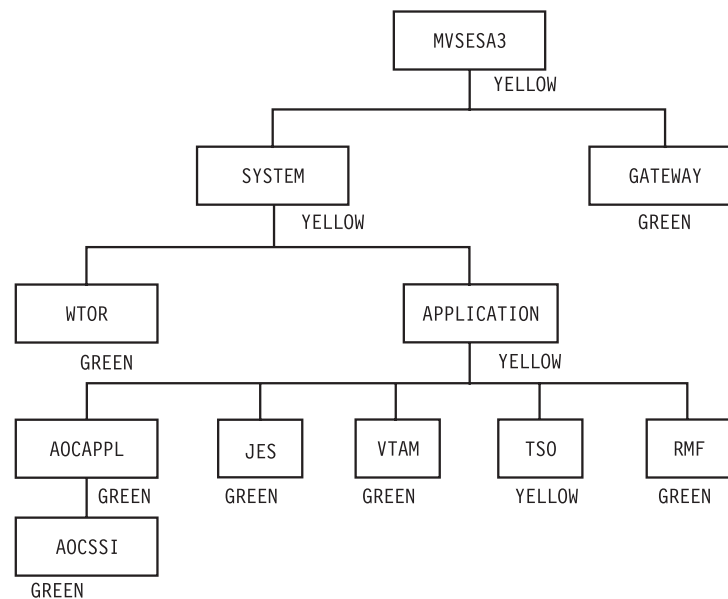
1. No automatic unlock. You must enter the Unlock command.
2. Automatic unlock when you press **F3** or **END** from the I/O operations ISPF main menu (IBM-supplied default).
3. Automatic unlock after every command.

6. Verify

Requires you to verify a command before the dialog activates it. Contrary to the other options, you can tailor the *Verify* option for the remainder of the session only. The default for this option resets to the IBM-supplied default at the beginning of each session. Selections are:

1. Verify before activating (IBM-supplied default).
 - If you press **Enter** after entering a connectivity command, the dialog will continue.
 - If you press **F3** or **END**, command processing is halted, and the preceding screen will be displayed.
2. Do not verify.

D. If You Select the Help Menu



1. How to get help...

- To get the entries shown above, select the *Help* menu from the I/O operations ISPF main menu.
- To get field-oriented help, press **F1** on any input field on a screen.

D. If You Select the Help Menu

- To get screen-oriented help, press **F1** on any full screen. If you press **F1** twice in a row, an ISPF help panel, not an I/O operations ISPF help panel, is displayed.
- 2. Help on keys... explains the I/O operations ISPF dialog settings of the following function keys: F1 (Help); F3 (End); F5 (Save); F6 (Activate).
- 3. Navigation Help... helps you navigate through the panel, or screen, flow of the I/O operations ISPF dialog.
- 4. Help Index... lets you choose among a selected number of topics that are listed alphabetically.
- 5. Tutorial ... Do one of the following:
 - Select the *Help* menu on the main menu. Then:
 - a. Select *Tutorial*
 - b. Select one of the topics shown in on the screen below.
 - Enter **TUTORIAL** on a panel of the dialog. Then:
 - a. Press **Enter**
 - b. Select one of the topics on the tutorial main menu shown below.

SA z/OS - IO-Ops ISPF Tutorial Main Menu

Select a specific topic by typing the topic number on the command line and pressing Enter.

```
1      Overview of IO-Ops and SWCHs
2      Displaying and Changing SWCH Configuration Data
3      Displaying Other Types of Information
4      Setting Options
5      Guidelines and Precautions
6      Understanding and Using the Matrix Screen
7      Making a Single Change to All SWCHs
8      Connectivity Command Options
```

COMMAND ==>

```
F1= Help      F2=          F3= END      F4= RETURN  F5=          F6=
F7= Help      F8=          F9=          PF10=       F11= Index  F12=
```

Appendix C. Problem Determination

This appendix contains various subtopics that deal with methods of avoiding or detecting problems during SA z/OS automation. The following topics are discussed:

- “Concurrent User Access within the Customization Dialog”
- “Maintaining System Status during a Status Forwarding Path Failure”
- “Debugging and Tracing” on page 274

Concurrent User Access within the Customization Dialog

The customization dialog allows you the editing of certain entry types by multiple users at the same time. To prevent simultaneous usage of the same resource, a serialization is implemented using the ENQ and DEQ macros with global resources. If an installation runs several systems in a global resource serialization (GRS) complex, but GRS is not installed, it should be ensured that the following major names (qname of ENQ resp. DEQ macro) are propagated:

INGPDB	This name is used by SA z/OS to serialize access within the customization dialog. It is mandatory to have this major name distributed among all systems where the customization dialog may run. Otherwise different users may overwrite their changes and data may get lost.
SYSDSN	This name is used by SA z/OS to obtain the user ID of a resource holder in case a policy database cannot be accessed. If this major name is not distributed, messages indicating that a resource cannot be accessed would not show a user ID.
SPFEDIT	This name is used by SA z/OS to obtain the user ID of a resource holder in case an ISPF table, which holds the data within a policy database, cannot be accessed. If this major name is not distributed, messages indicating that a resource cannot be accessed would not show a user ID.

Maintaining System Status during a Status Forwarding Path Failure

SA z/OS uses processor operations functions and XCF to attempt to maintain the current status of a system, even if there is a failure in the status forwarding path. SA z/OS system operations uses XCF for both passive and active monitoring of system status. As long as there is one SA z/OS in a sysplex with an active status forwarding path, SA z/OS is able to use XCF to maintain system status for all systems in the sysplex.

SA z/OS system operations receives alerts generated by processor operations for target systems monitored by processor operations. If, for example, a target system enters a wait state or a DCCF condition, SA z/OS will use the resulting alert from the processor operations functions to update the system status, even if the status forwarding path between SA z/OS and the target system is down.

Debugging and Tracing

This chapter describes the tracing and debugging facilities that are available with SA z/OS. The following topics are discussed:

- “Using Trace Services for the Automation Manager”
- “Using AOCTRACE” on page 276
- “How to Use a Log with INGHIST” on page 276
- “How to Use the Diagnostic Option of INGAMS” on page 277

For collecting debugging information you can also use the command INGLKUP REQ=COLLECT.

Using Trace Services for the Automation Manager

For tracing the control flow of a process, SA z/OS uses the **MVS Component Trace Facility** with all its capabilities.

This requires an External Writer as soon as the trace is started. The writer is internally started if it is not already active.

Before the trace can be activated the trace data set must be allocated. The recommended data set name is 'SYS1.HSA.sysname.CTRACE01'. Appropriate JCL is provided in the SINGSAMP SA z/OS sample library as member HSACTWR. In addition, the JCL of the component trace's external writer must be cataloged into SYS1.PROCLIB. For more information on customizing the component trace refer to *System Automation for z/OS Planning and Installation*.

To activate the trace, on an z/OS console, enter the command:

```
TRACE CT,ON,COMP=HSAAM
```

When the system responds with a WTOR expecting the trace options you reply:

```
xx,WTR=HSACTWR,OPTIONS=(ALL|set_of_options),END
```

where *set_of_options* means one of the following:

Option	Activates the trace in:
ALL	all of the following options
B[ase]	SA z/OS service routines using z/OS system services.
COM[munication]	the communications framework
CON[trol]	the base framework
E[vent]	the Sysplex Event Handler
F[ramework]	the service sustaining the LE environment
HE[althcheck]	the IBM Health Checker for z/OS and Sysplex
I[nstruction]	the services encoding or decoding the instruction streams
L[ock]	the service creating and deleting locks
MQ	WebSphere MQ services
R[equest]	the beginning and ending of a workitem (request) processing

Option	Activates the trace in:
S[chedule]	the service overriding service period definitions
W[orkitem]	all components working on work items

You deactivate the trace through the command:

```
TRACE CT,OFF,COMP=HSAAM
```

This command is automatically issued when the last automation manager in a single image terminates while the trace is active.

Before the trace records can be analyzed by IPCS the trace data set(s) must be made available to IPCS. This is done by the operator command:

```
TRACE CT,WTRSTOP=HSACTWR
```

If the related message indicates that more than one data set contains trace data, the IPCS **COPYTRC** command must be used to combine the data sets into a new one.

For analyzing the trace data the IPCS **CTRACE** command has to be used. JCL is provided for the IPCS batch interface. The command supports lots of parameters. At least the following must be adapted:

- COMP(HSAAM)
- USEREXIT(HSAPSTUE)
- DSNNAME(trace_data_set_name)
- OPTIONS((filter_options))

where **filter_options** is a combination the following items, separated by comma:

- BASE
- COMMUNICATION
- CONTROL
- EVENT
- FRAMEWORK
- HEALTHCHECK
- INSTRUCTION
- LOCK
- MQ
- REQUEST
- SCHEDULE
- WORKITEM
- AM=xcf_member_name
- CALLID=(callid[,callid])
- KEY=(key[,key])
- METHOD=(method[,method])
- MODULE=(module[,module])
- TASKID=(taskid[,taskid])
- TOKEN=(token[,token])

The keyword options are connected by a logical *AND* while the values of a keyword and the remaining options are connected by a logical *OR*. Except AM, each keyword option supports up to 5 different values separated by comma or space. Imbedded spaces or commas of a value require delimiters '..' or "..".

JCL Sample for Printing the Trace Records

The JCL sample member HSAJPTRC is stored in the SA z/OS library SINGSAMP. The data set names of the DDnames STEPLIB, IPCSDDIR, and TRACEDS must be changed to their actual values before the job is submitted. This is also true for the OPTIONS parameters completely listed because the values of the keywords are sample values only.

Using AOCTRACE

The AOCTRACE command can be used to enable or disable the automation debugging facility, either globally or for specific clists (REXX routines).

The syntax of the AOFTRACE command and additional information can be found in *System Automation for z/OS Operator's Commands*.

Information about using AOCTRACE to trace user-written automation procedures is provided in *System Automation for z/OS Customizing and Programming*.

For more information about trace settings, see TRACE in *TSO/E REXX/MVS Reference*.

How to Use a Log with INGHIST

With the INGHIST command you can specify the request to display a log with automation manager messages. For this purpose, type: INGHIST REQ=LOG on the command line to reach a panel similar to the one shown in Figure 143.

```

INGKYH11          SA z/OS - Command Dialogs          Line
Domain ID = IPUFM  ----- INGHIST -----          Date = 05/26/00
Operator ID = USER1      Sysplex = AOC7PLEX          Time = 11:41:12

Start time ==> 2000-05-26 10:41:00    format: yyyy-mm-dd hh:mm:ss
End time   ==>

Date          Time          Logger Messages          Group    XCF-Member
-----
2000-05-29 11:06:36      HSAM1308I SA z/OS PRIMARY AUTOMATION MANAGER INITIALIZATION
                                COMPLETE, TYPE=COLD.          INGXS   AOC7$$$1
2000-05-29 11:06:36      HSAM1311I Processing config part,
                                'SATEST.PROD.ACF.G0042V00(Z995CSCH)'.SCHEDULE.2000052314260
                                INGXS   AOC7$$$1
2000-05-29 11:06:36      HSAM1310I RC=2 REAS=4384 OPCODE=08020001 (MAIN.2)
                                ApplyConfiguration.          INGXS   AOC7$$$1

Command ==>
PF1=Help      PF2=End          PF3=Return          PF6=Roll
                PF9=Refresh          PF12=Retrieve
    
```

Figure 143. INGHIST Logger Messages

This panel displays automation manager messages that are written to the system logger. Two or more lines are displayed for each message depending on the size of the message text. The first line contains the date and time when the message was issued. It also shows the XCF member name of the automation manager and the XCF group name the automation manager belongs to. The second and following line(s) contain the message.

How to Use the Diagnostic Option of INGAMS

From the INGAMS command dialog, you can select option *D Diagnostic*. This leads you to the panel shown in Figure 144 where you can initiate the following diagnostic functions:

- Write snapshot of state image queue
- Start/stop recording

```

INGKYAM4                SA z/OS - Command Dialogs
Domain ID   = IPUFM      ----- INGAMS -----      Date = 05/26/00
Operator ID = USER1     Sysplex = AOCPLEX           Time = 11:09:32

Specify or revise the following data:

Action ==>  ___      1. Write Snapshot
                  2. Start recording
                  3. Stop recording

Data set name ==> _____

System       ==> _____ System name, domain id or sysplex name

Command ==>
PF1=Help    PF2=End    PF3=Return
                                           PF6=Roll
                                           PF12=Retrieve
  
```

Figure 144. Diagnostic

In this panel's fields, enter the following information:

Data set name

specify the name of the data set that will hold the snapshot data. The snapshot data set and the recording data set should be pre-allocated with RECFM FB and LRECL 80. The snapshot data set can be a sequential file or a member of a partitioned data set, however, the recording data set can only be a sequential file. It is strongly recommended that you use a partitioned data set extended (PDSE) or to make your allocation of the snapshot data set according to the snapshot size information that you will receive when selecting option *B Show Details* for an automation manager from the INGAMS command dialog. This displays the following panel showing the snapshot size (highlighted in the example).

```

INGKYAM3          SA z/OS - Command Dialogs          Line 1   of 30
Domain ID   = IPSFO   ----- INGAMS -----          Date = 05/27/04
Operator ID = USER1          Sysplex = KEY1PLEX          Time = 16:13:58

Automation Manager : KEY3$$$1          running on system : KEY3
Operation mode    : PAM                Job name         : AMA0
Status           : READY
XCF-Groupname    : INGXSGA0

Logic Deck ...
Release          : V2R3M0
Date built       : 15 May 2004
Time built      : 11:36:23
Last APAR       : NONE

Config dataset name : SATEST.SAMPLE.ACF
Config member      : HSAZ999
                  Z999CLGC  STRUCTURE  20040518085707
                  Z999CRES  STRUCTURE  20040526090342
                  Z999CSCH  SCHEDULE   20040526090342
                  Z999CRES  STRUCTURE  20040526090342
                  Z999CSCH  SCHEDULE   20040526090342
Config token      : 20040526090342000915282064
Config version    : 01

Diagnostic Info
Snapshot size      : 986132
Number of resources : 521
Number of requests  : 55
Number group requests : 49

-----Number of Messages-----
Queue                maximum current   high    low
HSA.INGXSGA0.WORKITEM.QUEUE  10000    0      8000   7000
HSA.INGXSGA0.AGENT.QUEUE    10000    0      8000   7000
HSA.INGXSGA0.STATE.QUEUE    50000    0     40000  35000

Command ==>
PF1=Help      PF2=End      PF3=Return   PF6=Ro11
              PF8=Forward  PF9=Refresh  PF12=Retrieve
    
```

Figure 145. INGAMS Command Dialog, Show Details

Note: A data set name is required when writing a snapshot or when starting recording.

System

is the name of the system where the automation manager resides. This is only necessary when the automation manager is not within the local sysplex. You can specify either the system name, the domain ID or the sysplex name.

Tracing and Debugging for ProcOps and the BCP Internal Interface

For ProcOps SNMP connections and the BCP Internal Interface trace facilities are provided that can be used for problem determination purposes.

tracing and debugging for ProcOps and the BCP Internal Interface

Using AOCTRACE

In order to enable the global tracing of all communications related parts of ProcOps-SNMP or BCP Internal Interface, issue the following command:

```
AOCTRACE compname ON
```

where *compname* is one of the following:

INGRX810

INGHWCMD command list

INGHWSND

routing interface module

INGHWCOM

communications module

To disable the global trace mode issue command:

```
AOCTRACE compname OFF
```

In global trace mode all currently active connections to SEs or HMCs are traced. This may produce a large amount of data written to the Netlog, depending on the number of connections or the amount event data returned from the hardware. In such cases, use the connection specific trace.

Tracing Specific Connections

If you want to trace the communication to a specific processor or CPC, issue the following service command:

```
INGHWCMD proname TRACE ON
```

where *proname* is the name of the target hardware or processor that is defined in the SA z/OS Customization Dialog.

Use the following service command to disable the trace:

```
INGHWCMD proname TRACE OFF
```

Although only a single processor connection is traced, the amount of data written to the Netlog may be high, especially if the processor connection is through an HMC. In this case, data from other processors serviced by this HMC is additionally shown in the trace. It is therefore recommended to limit the tracing time.

First Failure Data Capture Support

The INGHWCMD communications module supports the First Failure Data Capture debug aid, provided by NetView for high level language command processors, written in PL/I or C. In case of an INGHWCMD abnormal end, message CNM983E, CNM998E, or CNM999E is issued.

Message CNM983E contains the HLL API user trace information that is maintained by INGHWCMD. This information allows you to determine the internal program flow at error time. If available save this debug aid information before contacting IBM for service. For more information about the diagnostic messages, refer to the *Tivoli NetView for z/OS Diagnosis Guide*.

Debugging and Tracing

Appendix D. Additional BCP Internal Interfaces

Determine BCP Internal Interface Communication Task Names

The command 'INGHWCMD <pname> STATCOM' can be used to determine the name of the autotask for communicating with processor <pname>. In the AOFA0099 report string that is returned by this command, the TASK field contains the autotask name.

If the command cannot be applied, perform the following steps:

1. Issue command: QRYGLOBL COMMON VARS=ING.0HW*

A list of Cglobal variable name value pairs is displayed. Look for the processor name value whose communication autotask name you want to know, and make a note of the two digit variable name number <dd>. If the processor name is not shown in the list, communication to the processor has not been established.

2. Issue command: QRYGLOBL COMMON VARS=AOF.0HWOPER <dd>

A Cglobal variable name value pair is displayed with the autotask name stored as value.

BCP Internal Interface Recovery Actions

The following section explains manual recovery procedures that might be necessary in case the BCP Internal Interface communication module INGHWC.COM is not responding or is unavailable.

Interface report messages that indicate that module INGHWC.COM is unavailable are:

1. AOFA0000 Check Task: <tn>_task_module_INGHWC.COM_not_running
2. AOFA0000 Check Task: <tn>_reached QueueLimit_<ql>
3. AOFA0000 Check Task: <tn>_task_is_not_available

The recovery procedure for each of these cases is outlined below.

Recovery Case 1: <tn>_task_module_INGHWC.COM_not_running

The recovery procedure for this case is as follows:

1. Make a note of the autotask name <tn> and browse the netlog for messages to or from that task.
2. Search for DSI172I messages that have <tn> as the abending subtask and look for INGHWC.COM as the failing module. If this is the case, note down the last report message (AOFA*) prior to the abend from that task.
3. Issue the NetView command: EXCMD <tn> LOGOFF (where *tn* is the autotask name that you made a note of in step 1).

This will terminate the autotask and free its task storage that was allocated by INGHWC.COM. If the autotask has been defined as an HWOPERxx task in the SA z/OS AUTOOPS policy entry, it will be automatically restarted and activate the communication module INGHWC.COM.

Recovery Case 2: `<tn>_reached QueueLimit_<ql>`

Each Support Element connection uses a separate autotask. As soon as the "reached QueueLimit" message is returned, this indicates that the previously issued HW request has not yet sent a completion report. The normal recovery process would be to retry the HW command request at a later time.

There are, however, cases where module INGHWCOM has to wait for an indefinite period of time until the HW has completed the request. Usually, this occurs when a Support Element session has been started in SYNC processing mode while a long-running HW command such as ACTIVATE or CBU is still active

If necessary, you can interrupt and terminate the session, which will not affect any HW command currently in progress on the Support Element. If you terminate the session, you will not receive any completion information for this command.

Issue the following NetView commands:

1. EXCMD `<tn>` RESET
2. TASKUTIL `<tn>`
3. EXCMD `<tn>` LOGOFF

Command 1 terminates the communication module INGHWCOM in task `<tn>`. The session between NetView and the SE/HMC will be closed normally and the allocated task storage will be freed.

With command 2 you can control whether module INGHWCOM is no longer the active command of `<tn>` and the amount of storage still allocated by `<tn>`.

With command 3 you terminate autotask `<tn>`, which will initiate an autotask restart and the activation of INGHWCOM.

If command 2 indicates that command 1 did not end INGHWCOM, command 1 can be repeated with RESET IMMED. This will abnormally terminate INGHWCOM, but SE/HMC session termination will not be performed. Note that a 'STOP TASK' command will end the autotask, but will not invoke normal SE/HMC session termination.

Recovery Case 3: `<tn>_task_is_not_available`

There are cases where a BCP Internal Interface autotask, previously defined as an HWOPERxx task with an SA z/OS AUTOOPS policy entry, cannot be used. In these cases, a task definition problem or a mismatch of your current ACF configuration might be the reason for the problem.

The recovery procedure for this case is as follows:

1. Verify the AUTOOPS definition statements in your PDB.
2. Verify that the corresponding DSIOPF statements have AOFPRFHW defined as the operator profile name.
3. Perform a Build operation after the policy changes.
4. Activate your policy changes, for example by performing an ACF REFRESH.

Terminating NetView and the BCP Internal Interface

When terminating NetView, a 'CLOSE NORMAL' command should first be issued to give INGHWCOM enough time to end all its SE/HMC sessions. Then, after a predefined period of time, a 'CLOSE IMMED' command could be used to limit NetView's overall termination time.

BCP Internal Interface Recovery Actions

Appendix E. The IBM Health Checker for z/OS and Sysplex Checks

This appendix provides details of the checks carried out by the IBM Health Checker for z/OS and Sysplex.

Table 13 gives a list of the check names and indicates whether they are local or global, and what their interval is for repetitive checks:

Table 13. Overview of HealthChecker Best Practices Checks

Check name	Page	Type	Interval
ALTERNATE_CONSOLE_GROUPS	291	local	24 hours
AMRF_AND_MPF_CONSISTENT	292	local	24 hours
AVAILABLE_FRAME_QUEUE_THRESHOLDS	286	local	24 hours
CDS_DATASET_SEPARATION	287	global	1 hour
CONSOLE_MASTER	291	local	24 hours
CONSOLE_MSCOPE_AND_ROUTCODES	292	local	24 hours
CONSOLE_NAMES	290	local	24 hours
CONSOLE_ROUTCODE_11	292	local	24 hours
COUPLINGFACILITY_STRUCTURE	288	local	12 hours
EMCS_HARDCOPY	293	local	12 hours
EMCS_MSCOPE_AND_ROUTCODES	293	local	12 hours
GRS_MODE	297	global	24 hours
NUMBER_EMCS_CONSOLES	293	global	12 hours
REAL_STORAGE_AVAILABILITY	295	local	24 hours
RSU_STORAGE_AVAILABILITY	296	local	24 hours
SDUMP_AVAILABILITY	297	local	24 hours
SYS_CF_STR_REPORT	288	global	24 hours
SYSCONS_MASTER	295	local	8 hours
SYSCONS_MSCOPE	294	local	24 hours
SYSCONS_PD_MODE	295	local	24 hours
SYSCONS_ROUTCODES	294	local	24 hours
SYSPLEX_MASTER	291	global	24 hours
USS_FILESYS_CONFIG	286	local	24 hours
XCF_CLEANUP_VALUE	296	local	24 hours
XCF_FAILURE_DETECTION_INTERVAL	296	local	24 hours
XCF_SIGNALLING	289	local	12 hours
XCF_SIGNALLING_STRUCTURES_IN_CF	290	local	1 hour
XCF_SYSPLEX_FAILURE_MANAGEMENT	297	global	24 hours

The IBM Health Checker for z/OS and Sysplex Checks

Note: In the following list of checks, "User override" refers to your ability to specify parameters that override the IBM values. A subset of the checks support this. However, all checks can be individually disabled so that the check is not run.

- **Automove setup verification**

Check name: USS_FILESYS_CONFIG

Best practice: You should define your version and sysplex root HFS data as AUTOMOVE, and define your system-specific file systems as UNMOUNT. Do not define a file system as NOAUTOMOVE or UNMOUNT and a file system underneath it as AUTOMOVE. If you do, the file system defined as AUTOMOVE will not be available until the failing system is restarted. A sysplex file system that changes ownership as the result of a system failure, will only be accessible in the new environment if its mount point is also accessible. The Automove check verifies that your file systems are setup according to these rules. This check is only applicable for images that are part of a sysplex.

The AUTOMOVE|NOAUTOMOVE|UNMOUNT parameters on ROOT and MOUNT indicate what happens to the file system if the system that owns that file system goes down. The AUTOMOVE parameter specifies that ownership of the file system is automatically moved to another system. It is the default. The NOAUTOMOVE parameter specifies that the file system will not be moved if the owning system goes down and the file system is not accessible. – UNMOUNT specifies that the file system will be unmounted when the system leaves the sysplex.

User override: Yes

Parameters: For file system MODE, the parameter SYSPLEX ensures Automove support. For other file modes, the parameter NOPLEX will not check for Automove support.

Syntax:

```
"CHECK(USS_FILESYS_CONFIG)"
"SEVERITY(High)"
"DATE(20030102)"
"PARMS(SYSPLEX)"
"REASON('USS Automove moves a file system to a new system in
the Sysplex when the owning system fails');"
```

Reference: See *z/OS UNIX System Services Planning*, GA22-7800 and APAR II3129.

- **Available frame queue threshold, reclaiming storage frames**

Check name: Available_Frame_Queue_Thresholds

Best practice: To avoid situations where the system does not start to reclaim storage frames soon enough, you should evaluate the values for storage. If you are running in 31-bit mode, then both the MCCAFACTH and the MCCAECTH values are used. If you are running in 64-bit mode, then only the MCCAECTH value is used. For migrations to a 64-bit environment, this check is critical because the same value used in 31-bit mode could introduce problems. IBM suggests that the IEAOPTxx parameters are set as follows:

- MCCAFACTH

The IBM Health Checker for z/OS and Sysplex Checks

MCCAFCTH specifies the low and the OK threshold values for central storage. The *lowvalue* indicates the number of frames on the available frame queue when stealing begins. The *okvalue* indicates the number of frames on the available frame queue when stealing ends. You can monitor actual conditions on the RMF Paging Activity Report (RMF Monitor 1) or equivalent performance monitoring product and adjust accordingly.

– MCCAECTH

MCCAECTH specifies the low and the OK threshold values for expanded storage. The *lowvalue* indicates the number of frames on the available frame queue when real storage manager (RSM) frame stealing begins. The *okvalue* indicates the number of frames on the available frame queue when stealing ends. You can monitor actual conditions on the RMF Paging Activity Report (RMF Monitor 1) or equivalent performance monitoring product and adjust accordingly.

Note: This parameter is ignored in 64-bit mode. In 31-bit mode, the defaults are sufficient. For these two parameters, the defaults are MCCAECTH=(150,300). The OK point for available frames in a 31-bit mode implementation is 400 frames, 100 from central storage and 300 from expanded storage.

For 64-bit mode, after installing APARs OW55902 and OW55729, the default values for MCCAECTH are (400,600). These are IBM's minimum suggested settings. It is suggested that you allow MCCAECTH to default to (400,600). Higher values are acceptable.

User override: Yes

Parameters:

1. 64 bit Minimum LOW threshold (special action commences).
2. 64 bit Minimum OK threshold (special action ceases).
3. 31 bit Minimum LOW threshold (special action commences).
4. 31 bit Minimum OK threshold (special action ceases).

Syntax:

```
"CHECK(Available_Frame_Queue_Thresholds)"  
"SEVERITY(High)"  
"DATE(20030102)"  
"PARMS(400,600,200,400)"  
"REASON('System may not recover in time if set too low');"
```

Reference: See *z/OS MVS Initialization and Tuning Reference*, SA22-7592 for information about the MCCAECTH and MCCAECTH IEAOPTxx parameters, and *z/OS RMF Report Analysis*, SC33-7991 for information about using the Paging Activity report. You should also be familiar with the whitepaper *z/OS Performance: Managing Processor Storage in a 64-bit environment*, WP100269.

• Couple data set separation

Check name: CDS_Dataset_Separation

Best practice: There are three facets to this check:

The IBM Health Checker for z/OS and Sysplex Checks

- The primary sysplex, CFRM, and LOGR couple data sets should not reside on the same volume due to the amount of I/O activity for each of these data sets.
- For all couple data sets, the primary couple data sets should reside on a separate volume from the alternate couple data set.
- Each primary couple data set has an active alternate couple data set.

User override: No

Parameters: Not applicable.

Syntax:

```
"CHECK(CDS_Dataset_Separation)"  
"SEVERITY(High)"  
"DATE(20030102)"  
"REASON('Ensure that CDS separation has been maintained');"
```

Reference: The publication, *Parallel Sysplex® Availability Checklist*, provides detailed recommendations and characteristics about the placement of primary and alternate couple data sets. See also the section, "Planning for the couple data sets," in *z/OS MVS Setting Up a Sysplex*, SA22-7625.

• Coupling facility structure attributes and location

Check name: CouplingFacility_Structure

Best practice: This check also displays each of the defined coupling facilities, their status, and the relationship between the coupling facilities and structures. The check compares placement based on the preference list, specified in the CFRM policy, which is used to designate the location of coupling facility structures for performance and capacity considerations.

This check shows current status and attributes of each coupling facility. For example, it shows whether a coupling facility is volatile or nonvolatile. Determine if the current status differs from your expectations or requirements.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(CouplingFacility_Structure)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"REASON('Check CF and Structure location');"
```

Reference: Refer to the following sections in *z/OS MVS Setting Up a Sysplex*, SA22-7625: "Understanding preference and exclusion lists" for information about specification of preferences; "Using the POPULATECF Function to Rebuild Coupling Facility Structures" if you want to rebuild any of the coupling facility structures in their preferred coupling facility.

• Create report for coupling facilities, structures, and systems

Check name: Sys_CF_STR_Report

Best practice: This check produces a report displaying systems, coupling facilities, structures and status of these resources.

The IBM Health Checker for z/OS and Sysplex Checks

User override: No

Parameters: None required.

Syntax:

```
"CHECK(Sys_CF_STR_Report) "  
"SEVERITY(Low)"  
"DATE(20030102)"  
"REASON('Create System, CF, Structure report');"
```

Reference: See the topic about Sysplex policies in *Parallel Sysplex Availability Checklist* for recommendations about structure placement, preferences, and characteristics of structures.

• Cross system coupling facility (XCF) signalling

Check name: XCF_Signalling

Best practice: This check verifies the following:

1. all transport classes should be set up to service the pseudo-group name 'UNDESIG'. This ensures that any XCF message can use each transport class.
2. all defined Transport Classes are assigned to at least one pathout (outbound path).
3. most pathouts have a transport class defined with a "small" (parameter #4) classlength, and at least one other transport class is defined with a higher "large" (parameter #5) classlength.
4. multiple pathout/pathin pairs (at least parameter #3) are in the operational state for each system in the sysplex that is connected to the current system.
5. a MAXMSG value of a minimum size (parameter #1) is defined for each transport class.
6. each inbound signal path can support a minimum number of messages (parameter #2) from the sending system.

These actions avoid a single point of failure. Exception conditions flagged by this check can reflect a hardware or configuration problem.

User override: Yes

Parameters: Check #1: None required.

Check #2: None required.

Check #3: parameter 4: Specifies the maximum value to be interpreted as a "small" (XCF transport) classlength. This is an *integer*. The maximum acceptable value is 9999. The specified value does not include the 68 additional bytes used by XCF for internal control blocks.

Check #3 parameter 5: Specifies the minimum value to be interpreted as a "large" (XCF transport) classlength. This is an *integer*. The maximum acceptable value is 62464. The minimum acceptable value is 4028. The specified value does not include the 68 additional bytes used by XCF for internal control blocks

Check #4 parameter 3: Specifies the minimum pathout/pathin pair count for a system. This is an *integer*. The maximum acceptable value is 9.

The IBM Health Checker for z/OS and Sysplex Checks

Check #5 parameter 1: The minimum MAXMSG value for transport classes. This is an *integer*. The maximum acceptable value is 999999.

Check #6 parameter 2: The minimum number of XCF messages that an inbound XCF signal path should support to avoid message backup. This is an *integer*. The maximum acceptable value is 999999.

Syntax:

```
"CHECK(XCF_Signalling)"
"SEVERITY(Low)"
"DATE(20030102)"
"PARMS(750,30,2,956,4028)"
"REASON('Avoid problems with XCF signalling.');
```

Reference: See the topic about Sysplex policies in *Parallel Sysplex Availability Checklist* for recommendations about structure placement, preferences, and characteristics of structures.

- **Cross system coupling facility (XCF) structure location**

Check name: XCF_Signalling_Structures_in_CF

Best practice: If multiple XCF signaling structures are in use, then all of them should not reside on the same coupling facility. There should be at least two online, operational links to each coupling facility. Also, there should not be fewer operational links (CHPID) than there are active links. These actions avoid a single point of failure. Conditions flagged by this check can reflect a hardware problem.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(XCF_Signalling_Structures_in_CF)"
"SEVERITY(Medium)"
"DATE(20030102)"
"REASON('Avoid problems with XCF signalling in CFs.');
```

Reference: See the topic about Sysplex policies in *Parallel Sysplex Availability Checklist* for recommendations about structure placement, preferences, and characteristics of structures.

- **Sysplex console checks**

The following group of checks is performed:

- Consoles are assigned names.

Check name: Console_Names

Best practice: IBM suggests that MCS, SNA_MCS, and subsystem consoles are assigned names; this reduces the number of console IDs to help address the limit of 99 consoles per sysplex. Console names are specified within the CONSOLxx parmlib entry, using the NAME parameter. The assignment of names to consoles is also required to use alternate groups for consoles.

User override: No

Parameters: None required.

The IBM Health Checker for z/OS and Sysplex Checks

Syntax:

```
"CHECK(Console_Names)"  
"SEVERITY(High)"  
"DATE(20030102)"  
"REASON('Like named consoles are matched across the sysplex');"
```

- Alternate groups are defined for consoles.

Check name: Alternate_Console_groups

Best practice: IBM suggests that you define alternate groups for consoles (using the ALTGRP parameter of the CONSOLxx parmlib member). This increases availability if there is a console failure. In such cases, MVS will attempt to switch to another console. Specifying alternate groups (ALTGRP) is preferable to the use of alternate consoles (ALTCONS).

User override: No

Parameters: None required.

Syntax:

```
"CHECK(Alternate_Console_groups)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"REASON('Provides good recovery from console loss');"
```

- Consoles on each system have a console with master authority that has been defined with command association.

Check name: Console_master

Best practice: IBM suggests there is a console defined with both MASTER authority and command association for each system in the sysplex.

For the Console_master check to be successful, each system in the sysplex requires a console. If you did not configure your sysplex so that each system has a console, then you should consider disabling this check using the NOCALL parameter. IBM requests feedback on the value of this check.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(Console_master)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"REASON('Needed for DCCF and other situations');"
```

- Master console is active

Check name: Sysplex_master

Best practice: IBM suggests that the Sysplex Master Console is active within the sysplex.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(Sysplex_master)"  
"SEVERITY(High)"  
"DATE(20030102)"  
"REASON('Needed in emergencies');"
```

The IBM Health Checker for z/OS and Sysplex Checks

- Console message scope and routing codes

Check name: Console_MSCOPE_and_Routcodes

Best practice: Due to the potentially high volume of messages that could be received by a console, IBM suggests that consoles limit the messages and routing codes received to that console's functions. This will improve availability by improving performance and preventing buffer shortages. For example, a console that has a multisystem scope, should receive messages specific to that console's function. Conversely, consoles that are configured ROUTCODE(ALL) should limit the scope of messages received to a single system.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(Console_MSCOPE_and_Routcodes)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"REASON('Avoids overloading any console. Reduces number of  
messages sent to Sysplex consoles');"
```

- Use of Action message retention facility (AMRF) and retention of eventual action messages

Check name: AMRF_and_MPF_consistent

Best practice: IBM performs this check only if you are using AMRF. The messages can be retrieved at a later time (using the DISPLAY R command). Also, eventual action messages should not be retained to keep the message from becoming too long.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(AMRF_and_MPF_consistent)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"REASON('Avoids long chains of messages in storage');"
```

- No console is receiving route code 11 messages

Check name: Console_routcode_11

Best practice: Operator consoles do not need to receive routing code 11, which are system programmer messages. This keeps unnecessary messages from being delivered to a console. Route code 11 messages can be retrieved using the DISPLAY R,CE command.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(Console_routcode_11)"  
"SEVERITY(Low)"  
"DATE(20030102)"  
"REASON('Not really needed as for programmer info only');"
```

Reference: See *z/OS MVS Planning: Operations*, SA22-7601; the Consoles

The IBM Health Checker for z/OS and Sysplex Checks

topic in *Parallel Sysplex Availability Checklist*; and *Parallel Sysplex Managing Software for Availability*, SG24-5451.

- **Extended master console (EMCS) checks**

To extend the number of consoles on MVS systems, or to allow applications and programs to access MVS messages and send commands, an installation can use extended MCS consoles. The use of these consoles can help alleviate the constraint of the 99 MCS console limit. Moving to an extended MCS console base from a subsystem-allocatable console base will allow for easier expansion in a sysplex.

Once an EMCS console is defined and activated, it lives for the life of the sysplex-whether it remains active or not. After the number of EMCS consoles (including inactive consoles) becomes very large, console initialization during IPL can be elongated by minutes. This may occur due to an error in NetView setup or if a CLIST does not reuse EMCS console names. This results in an on-going increase in the number of EMCS consoles defined.

- Extended master console messages scope and routing codes

Check name: EMCS_Mscope_and_Routcode

Best practice: Due to the potentially high volume of messages that could be received by a console, IBM suggests that EMCS consoles limit the messages and routing codes received to that console's functions. This will improve availability by improving performance and preventing buffer shortages. For example, if an EMCS console is receiving messages from multiple systems, limit the number of route codes assigned to this console. You should not specify ROUTCODE(ALL). Conversely, if an EMCS console is intended to receive all route codes, the scope should be limited to a single system.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(EMCS_Mscope_and_Routcode)"
"SEVERITY(Medium)"
"DATE(20030102)"
"REASON('ROUTCODE(ALL) and non-local MSCOPE will cause a large
number of messages to be processed');"
```

- Extended consoles with master authority should not be allowed to receive hardcopy messages or to be backup devices for hardcopy medium.

Check name: EMCS_hardcopy

Best practice: An EMCS console should not be defined to receive hardcopy messages if the message scope (MSCOPE) is greater than 1.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(EMCS_hardcopy)"
"SEVERITY(Medium)"
"DATE(20030102)"
"REASON('EMCS consoles with HARDCOPY specified will process an
excessive number of messages');"
```

- Number of EMCS consoles is within recommended range

Check name: Number_EMCS_consoles

The IBM Health Checker for z/OS and Sysplex Checks

Best practice: If the combined total of active and inactive EMCS consoles is excessive, performance can be impacted.

User override: Yes

Parameters:

1. Maximum number of *active* EMCS consoles on this system. Values between 0 and 99999999 are accepted. Must be numeric.
2. Maximum number of *inactive* EMCS consoles on the entire sysplex. Values between 0 and 99999999 are accepted. Must be numeric.

Syntax:

```
"CHECK(Number_EMCS_consoles)"  
"SEVERITY(High)"  
"DATE(20030102)"  
"PARMS(5000,10000)"  
"REASON('Excessive numbers of EMCS consoles cause slowdown');"
```

Reference: See: *z/OS MVS Planning: Operations*, SA22-7601; the Consoles topic in *Parallel Sysplex Availability Checklist*; and *Parallel Sysplex Managing Software for Availability*, SG24-5451.

- **MVS system console checks**

The following group of checks is performed:

- System console is defined to have a local message scope

Check name: SYSCONS_MSCOPE

Best practice: MVS system consoles should be defined to have a local message scope. This reduces the amount of message traffic and improves performance and availability. This is of particular importance when the MVS system console is used during recovery actions.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(SYSCONS_MSCOPE)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"REASON('If SYSCONS is used in emergencies it should not have to process large numbers of messages');"
```

- System console is defined to have a limited number of routing codes

Check name: SYSCONS_ROUTCODES

Best practice: MVS system consoles should be defined to have a limited set of routing codes. This reduces the amount of message traffic and improves performance and availability. The console should be defined with either ROUTCODE(1,2,10) or ROUTCODE(NONE). This is of particular importance when the MVS system console is used during recovery actions.

User override: No

Parameters: None required.

The IBM Health Checker for z/OS and Sysplex Checks

Syntax:

```
"CHECK(SYSCONS_ROUTCODES) "  
"SEVERITY(Low)"  
"DATE(20030102)"  
"REASON('If SYSCONS is used in emergencies it should not have  
to process large numbers of messages');"
```

- System consoles are not running in problem determination mode

Check name: SYSCONS_PD_MODE

Best practice: System consoles should not be running in problem determination mode during normal operations. Problem determination mode degrades performance.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(SYSCONS_PD_MODE) "  
"SEVERITY(Low)"  
"DATE(20030102)"  
"REASON('SYSCONS should be run in Problem Determination mode  
only when there is a problem');"
```

- Active system console is defined with MASTER authority

Check name: SYSCONS_MASTER

Best practice: The active MCS system console should be defined to have MASTER authority. This is of particular importance when the MVS system console is used as a backup to the sysplex master console.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(SYSCONS_MASTER) "  
"SEVERITY(High)"  
"DATE(20030102)"  
"REASON('SYSCONS needs MASTER authority to resolve problems in  
emergency situations');"
```

Reference: See: *z/OS MVS Planning: Operations*, SA22-7601; *Consoles topic in Parallel Sysplex Availability Checklist*; and *Parallel Sysplex Managing Software for Availability*, SG24-5451.

• Real storage settings

Check name: Real_Storage_Availability

Best practice: IBM suggests that both the real and reconfigurable storage parameters should be set to 0. However, this would not be valid if you need to reconfigure storage or to run V=R jobs. The IEASYSxx parmlib member should specify the REAL parameter as REAL=0. This will improve performance.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(Real_Storage_Availability) "  
"SEVERITY(Low)"  
"DATE(20030102)"  
"REASON('Performance may be impacted');"
```

The IBM Health Checker for z/OS and Sysplex Checks

Reference: See *z/OS MVS Initialization and Tuning Reference*, SA22-7592.

- **Reconfigurable storage settings**

Check name: RSU_Storage_Availability

Best practice: IBM suggests that both the real and reconfigurable storage parameters should be set to 0. RSU reflects the amount of central storage to be made available for storage reconfiguration. The IEASYSxx parmlib member should specify the RSU parameter as RSU=0.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(RSU_Storage_Availability)"
"SEVERITY(Low)"
"DATE(20030102)"
"REASON('Performance may be impacted');"
```

Reference: See *z/OS MVS Initialization and Tuning Reference*, SA22-7592.

- **XCF cleanup value**

Check name: XCF_Cleanup_Value

Best practice: You should specify a value of 15 for the CLEANUP parameter in the COUPLExx parmlib member. Cleanup specifies how many seconds the system waits for before notifying members that this system is terminating, and loading a nonrestartable wait state. This is the amount of time that members of the sysplex have to perform cleanup processing.

User override: Yes

Parameters: Recommended XCF cleanup time in seconds. The maximum acceptable value is 86400.

Syntax:

```
"CHECK(XCF_Cleanup_Value)"
"SEVERITY(Low)"
"DATE(20030102)"
"PARMS(15)"
"REASON('Quick removal of a dead system from SYSPLEX');"
```

Reference: See *z/OS MVS Initialization and Tuning Reference*, SA22-7592.

- **Sysplex failure detection interval**

Check name: XCF_Failure_Detection_Interval

Best practice: The CLEANUP INTERVAL parameter in the COUPLE xx parmlib member must be coordinated with the spin recovery actions (SPINRCVY) statement in the EXSPATxx parmlib member. The HealthChecker checks that the CLEANUP INTERVAL value conforms to the IBM formula (2*SPINRCVT+5).

The spintime should be defined as 10 seconds for a system in either basic mode or LPAR mode with dedicated CPs. The spintime should be defined as 40 seconds for LPAR mode when the CPs are shared.

User override: Yes

Parameters: PARM1 and PARM2 in the formula PARM1*SPINRCVT+PARM2.

The IBM Health Checker for z/OS and Sysplex Checks

Syntax:

```
"CHECK(XCF_Failure_Detection_Interval)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"PARMS(2,5)"  
"REASON('Allow adequate time to recover from spin situation  
before system is assumed dead');"
```

Reference: For information about the EXSPATxx parmlib member, refer to *z/OS MVS Initialization and Tuning Reference*, SA22-7592 and to *z/OS MVS Setting Up a Sysplex*, SA22-7625.

• Sysplex failure management (SFM) is active

Check name: XCF_SYSPLEX_Failure_Management

Best practice: IBM suggests that you use sysplex failure management (SFM) to define actions to be performed in the event of:

- Signaling connectivity failures in the sysplex
- System failures, indicated by a status update missing condition
- The need to reconfigure systems in a PR/SM environment.

User override: Yes

Parameters: Recommended SFM status (ACTIVE/INACTIVE).

Syntax:

```
"CHECK(XCF_SYSPLEX_Failure_Management)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"PARMS(ACTIVE)"  
"REASON('An SFM policy provides better failure management');"
```

Reference: See *z/OS MVS Setting Up a Sysplex*, SA22-7625 for information about defining SFM policies and the *Parallel Sysplex Availability Checklist*.

• SVC- dump is using dynamically allocated data sets

Check name: SDUMP_Availability

Best practice: IBM suggests that you use dynamic allocation for your dump data sets to ensure that complete diagnostic data is captured at the first occurrence. If your dump data sets are not dynamically allocated and become full, you can lose important diagnostic information.

User override: No

Parameters: None required.

Syntax:

```
"CHECK(SDUMP_Availability)"  
"SEVERITY(Medium)"  
"DATE(20030102)"  
"REASON('SDUMP setup should ensure adequate diagnostics are  
gathered on the 1st occurrence of problems');"
```

Reference: See *z/OS MVS Diagnosis: Tools and Service Aids*, GA22-7589 for information about allocating stand-alone dump data sets.

• Global Resource Serialization (GRS) STAR configuration

Check name: GRS_Mode

Best practice: A STAR configuration is recommended due to the advantages

The IBM Health Checker for z/OS and Sysplex Checks

that it provides with regard to availability, real storage consumption, processing capacity, and response time.

User override: Yes

Parameters: Mode required, STAR, RING or NONE.

Syntax:

```
"CHECK(GRS_Mode)"  
"SEVERITY(High)"  
"DATE(20030102)"  
"PARMS(STAR)"  
"REASON('GRS should run in STAR mode to improve performance.');
```

Reference: See *z/OS MVS Planning: Global Resource Serialization*, SA22-7600.

Glossary

This glossary includes terms and definitions from:

- The *IBM Dictionary of Computing* New York: McGraw-Hill, 1994.
- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies can be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018. Definitions are identified by the symbol (A) after the definition.
- The *Information Technology Vocabulary* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.

The following cross-references are used in this glossary:

Contrast with. This refers to a term that has an opposed or substantively different meaning.

Deprecated term for. This indicates that the term should not be used. It refers to a preferred term, which is defined in its proper place in the glossary.

See. This refers the reader to multiple-word terms in which this term appears.

See also. This refers the reader to terms that have a related, but not synonymous, meaning.

Synonym for. This indicates that the term has the same meaning as a preferred term, which is defined in the glossary.

Synonymous with. This is a backward reference from a defined term to all other terms that have the same meaning.

A

ACF. Automation control file.

ACF/NCP. Advanced Communications Function for the Network Control Program. See *Advanced Communications Function* and *Network Control Program*.

ACF/VTAM. Advanced Communications Function for the Virtual Telecommunications Access Method. Synonym for *VTAM*. See *Advanced Communications Function* and *Virtual Telecommunications Access Method*.

active monitoring. In SA z/OS, the acquiring of resource status information by soliciting such information at regular, user-defined intervals. See also *passive monitoring*.

adapter. Hardware card that enables a device, such as a workstation, to communicate with another device, such as a monitor, a printer, or some other I/O device.

Address Space Workflow. In RMF, a measure of how a job uses system resources and the speed at which the job moves through the system. A low workflow indicates that a job has few of the resources it needs and is contending with other jobs for system resources. A high workflow indicates that a job has all the resources it needs to execute.

adjacent hosts. Systems connected in a peer relationship using adjacent NetView sessions for purposes of monitoring and control.

adjacent NetView. In SA z/OS, the system defined as the communication path between two SA z/OS systems that do not have a direct link. An adjacent NetView is used for message forwarding and as a communication link between two SA z/OS systems. For example, the adjacent NetView is used when sending responses from a focal point to a remote system.

Advanced Communications Function (ACF). A group of IBM licensed programs (principally VTAM, TCAM, NCP, and SSP) that use the concepts of Systems Network Architecture (SNA), including distribution of function and resource sharing.

advanced program-to-program communication (APPC). A set of inter-program communication services that support cooperative transaction processing in a Systems Network Architecture (SNA) network. APPC is the implementation, on a given system, of SNA's logical unit type 6.2.

alert. (1) In SNA, a record sent to a system problem management focal point or to a collection point to communicate the existence of an alert condition. (2) In NetView, a high-priority event that warrants immediate

attention. A database record is generated for certain event types that are defined by user-constructed filters.

alert condition. A problem or impending problem for which some or all of the process of problem determination, diagnosis, and resolution is expected to require action at a control point.

alert focal-point system. See entry for NPDA focal-point system under *focal—point system*.

alert threshold. An application or volume service value that determines the level at which SA z/OS changes the associated icon in the graphic interface to the alert color. SA z/OS may also issue an alert. See *warning threshold*.

AMC. (1) Automation Manager Configuration (2) The Auto Msg Classes entry type

APF. Authorized program facility.

API. Application programming interface.

APPC. Advanced program-to-program communications.

application. An z/OS subsystem or job monitored by SA z/OS.

Application entry. A construct, created with the customization dialogs, used to represent and contain policy for an application.

application group. A named set of applications. An application group is part of an SA z/OS enterprise definition and is used for monitoring purposes.

ApplicationGroup entry. A construct, created with the customization dialogs, used to represent and contain policy for an application group.

application program. (1) A program written for or by a user that applies to the user's work, such as a program that does inventory or payroll. (2) A program used to connect and communicate with stations in a network, enabling users to perform application-oriented activities.

ARM. Automatic restart management.

ASCB. Address space control block.

ASCB status. An application status derived by SA z/OS running a routine (the ASCB checker) that searches the z/OS address space control blocks (ASCBs) for address spaces with a particular job name. The job name used by the ASCB checker is the job name defined in the customization dialog for the application.

ASCII (American National Standard Code for Information Interchange). The standard code, using a coded character set consisting of 7-bit coded characters

(8-bit including parity check), for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters. (A)

ASF. Automation status file.

assist mode facility. An SA z/OS facility that uses SDF and enables interaction with automation before SA z/OS takes an automation action. SDF prompts the operator with a suggested action, then provides options for using that action, modifying and using the action, or canceling the action. Also called assist mode, it is enabled using the customization dialogs, or dynamically.

authorized program facility (APF). A facility that permits identification of programs that are authorized to use restricted functions.

automated function. SA z/OS automated functions are automation operators, NetView autotasks that are assigned to perform specific automation functions. However, SA z/OS defines its own synonyms, or *automated function names*, for the NetView autotasks, and these function names are referred to in the sample policy databases provided by SA z/OS. For example, the automation operator AUTBASE corresponds to the SA z/OS automated function BASEOPER.

automated console operations (ACO). The concept (versus a product) of using computers to perform a large subset of tasks ordinarily performed by operators, or assisting operators in performing these tasks.

automatic restart management. A z/OS recovery function that improves the availability of specified subsystems and applications by automatically restarting them under certain circumstances. Automatic restart management is a function of the Cross-System Coupling Facility (XCF) component of z/OS.

automatic restart management element name. In MVS 5.2 or later, z/OS automatic restart management requires the specification of a unique sixteen character name for each address space that registers with it. All automatic restart management policy is defined in terms of the element name, including SA z/OS's interface with it.

automation. The automatic initiation of actions in response to detected conditions or events. SA z/OS provides automation for z/OS applications, z/OS components, and remote systems that run z/OS. SA z/OS also provides tools that can be used to develop additional automation.

automation agent. In SA z/OS, the automation function is split up between the automation manager and the automation agents. The observing, reacting and doing parts are located within the NetView address

space, and are known as the *automation agents*. The automation agents are responsible for:

- recovery processing
- message processing
- active monitoring: they propagate status changes to the automation manager

| **automation configuration file.** The data set that
| consists of:

- | • the automation control file (ACF)
- | • the automation manager configuration file (AMC)
- | • the NetView automation table (AT)
- | • the MPFLSTSA member

automation control file (ACF). In SA z/OS, a file that contains system-level automation policy information. There is one master automation control file for each NetView system on which SA z/OS is installed. Additional policy information and all resource status information is contained in the policy database (PDB). The SA z/OS customization dialogs must be used to build the automation control files. They must not be edited manually.

automation flags. In SA z/OS, the automation policy settings that determine the operator functions that are automated for a resource and the times during which automation is active. When SA z/OS is running, automation is controlled by automation flag policy settings and override settings (if any) entered by the operator. Automation flags are set using the customization dialogs.

automation manager. In SA z/OS, the automation function is split up between the automation manager and the automation agents. The coordination, decision making and controlling functions are processed by each sysplex's *automation manager*.

The automation manager contains a model of all of the automated resources within the sysplex. The automation agents feed the automation manager with status information and perform the actions that the automation manager tells them to.

The automation manager provides *sysplex-wide* automation.

Automation Manager Configuration. The Automation Manager Configuration file (AMC) contains an image of the automated systems in a sysplex or of a standalone system.

Automation NetView. In SA z/OS the NetView that performs routine operator tasks with command procedures or uses other ways of automating system and network management, issuing automatic responses to messages and management services units.

automation operator. NetView automation operators are NetView autotasks that are assigned to perform specific automation functions. See also *automated*

function. NetView automation operators may receive messages and process automation procedures. There are no logged-on users associated with automation operators. Each automation operator is an operating system task and runs concurrently with other NetView tasks. An automation operator could be set up to handle JES2 messages that schedule automation procedures, and an automation statement could route such messages to the automation operator. Similar to *operator station task*. SA z/OS message monitor tasks and target control tasks are automation operators.

automation policy. The policy information governing automation for individual systems. This includes automation for applications, z/OS subsystems, z/OS data sets, and z/OS components.

automation policy settings. The automation policy information contained in the automation control file. This information is entered using the customization dialogs. You can display or modify these settings using the customization dialogs.

automation procedure. A sequence of commands, packaged as a NetView command list or a command processor written in a high-level language. An automation procedure performs automation functions and runs under NetView.

automation status file. In SA z/OS, a file containing status information for each automated subsystem, component or data set. This information is used by SA z/OS automation when taking action or when determining what action to take. In Release 2 and above of AOC/MVS, status information is also maintained in the operational information base.

automation table (AT). See *NetView automation table*.

autotask. A NetView automation task that receives messages and processes automation procedures. There are no logged-on users associated with autotasks. Each autotask is an operating system task and runs concurrently with other NetView tasks. An autotask could be set up to handle JES2 messages that schedule automation procedures, and an automation statement could route such messages to the autotasks. Similar to *operator station task*. SA z/OS message monitor tasks and target control tasks are autotasks. Also called *automation operator*.

available. In VTAM programs, pertaining to a logical unit that is active, connected, enabled, and not at its session limit.

B

basic mode. A central processor mode that does not use logical partitioning. Contrast with *logically partitioned (LPAR) mode*.

BCP Internal Interface. Processor function of CMOS-390, zSeries processor families. It allows the communication between basic control programs such as z/OS and the processor support element in order to exchange information or to perform processor control functions. Programs using this function can perform hardware operations such as ACTIVATE or SYSTEM RESET.

beaconing. The repeated transmission of a frame or messages (beacon) by a console or workstation upon detection of a line break or outage.

BookManager. An IBM product that lets users view softcopy documents on their workstations.

C

central processor (CP). The part of the computer that contains the sequencing and processing facilities for instruction execution, initial program load (IPL), and other machine operations.

central processor complex (CPC). A physical collection of hardware that consists of central storage, one or more central processors, timers, and channels.

central site. In a distributed data processing network, the central site is usually defined as the focal point for alerts, application design, and remote system management tasks such as problem management.

CFR/CFS and ISC/ISR. I/O operations can display and return data about integrated system channels (ISC) connected to a coupling facility and coupling facility receiver (CFR) channels and coupling facility sender (CFS) channels.

channel. A path along which signals can be sent; for example, data channel, output channel. See also *link*.

channel path identifier. A system-unique value assigned to each channel path.

CHPID. In SA z/OS, channel path ID; the address of a channel.

CHPID port. A label that describes the system name, logical partitions, and channel paths.

channel-attached. (1) Attached directly by I/O channels to a host processor (for example, a channel-attached device). (2) Attached to a controlling unit by cables, rather than by telecommunication lines. Contrast with *link-attached*. Synonymous with *local*.

CI. Console integration.

CICS/VS. Customer Information Control System for Virtual Storage.

CLIST. Command list.

clone. A set of definitions for application instances that are derived from a basic application definition by substituting a number of different system-specific values into the basic definition.

clone ID. A generic means of handling system-specific values such as the MVS SYSCLONE or the VTAM subarea number. Clone IDs can be substituted into application definitions and commands to customize a basic application definition for the system that it is to be instantiated on.

CNC. A channel path that transfers data between a host system image and an ESCON control unit. It can be point-to-point or switchable.

command. A request for the performance of an operation or the execution of a particular program.

command facility. The component of NetView that is a base for command processors that can monitor, control, automate, and improve the operation of a network. The successor to NCCF.

command list (CLIST). (1) A list of commands and statements, written in the NetView command list language or the REXX language, designed to perform a specific function for the user. In its simplest form, a command list is a list of commands. More complex command lists incorporate variable substitution and conditional logic, making the command list more like a conventional program. Command lists are typically interpreted rather than being compiled. (2) In SA z/OS, REXX command lists that can be used for automation procedures.

command procedure. In NetView, either a command list or a command processor.

command processor. A module designed to perform a specific function. Command processors, which can be written in assembler or a high-level language (HLL), are issued as commands.

Command Tree/2. An OS/2-based program that helps you build commands on an OS/2 window, then routes the commands to the destination you specify (such as a 3270 session, a file, a command line, or an application program). It provides the capability for operators to build commands and route them to a specified destination.

common commands. The SA z/OS subset of the CPC operations management commands.

common routine. One of several SA z/OS programs that perform frequently used automation functions. Common routines can be used to create new automation procedures.

Common User Access (CUA) architecture. Guidelines for the dialog between a human and a workstation or terminal.

communication controller. A type of communication control unit whose operations are controlled by one or more programs stored and executed in the unit or by a program executed in a processor to which the controller is connected. It manages the details of line control and the routing of data through a network.

communication line. Deprecated term for *telecommunication line*.

connectivity view. In SA z/OS, a display that uses graphic images for I/O devices and lines to show how they are connected.

console automation. The process of having NetView facilities provide the console input usually handled by the operator.

console connection. In SA z/OS, the 3270 or ASCII (serial) connection between a PS/2 computer and a target system. Through this connection, the workstation appears (to the target system) to be a console.

console integration (CI). A hardware facility that if supported by an operating system, allows operating system messages to be transferred through an internal hardware interface for display on a system console. Conversely, it allows operating system commands entered at a system console to be transferred through an internal hardware interface to the operating system for processing.

consoles. Workstations and 3270-type devices that manage your enterprise.

Control units. Hardware units that control I/O operations for one or more devices. You can view information about control units through I/O operations, and can start or stop data going to them by blocking and unblocking ports.

controller. A unit that controls I/O operations for one or more devices.

couple data set. A data set that is created through the XCF couple data set format utility and, depending on its designated type, is shared by some or all of the z/OS systems in a sysplex. See also *sysplex couple data set* and *XCF couple data set*.

coupling facility. The hardware element that provides high-speed caching, list processing, and locking functions in a sysplex.

CP. Central processor.

CPC. Central processor complex.

CPC operations management commands. A set of commands and responses for controlling the operation of System/390 CPCs.

CPC subset. All or part of a CPC. It contains the minimum *resource* to support a single control program.

CPCB. Command processor control block; an I/O operations internal control block that contains information about the command being processed.

CPU. Central processing unit. Deprecated term for *processor*.

cross-system coupling facility (XCF). XCF is a component of z/OS that provides functions to support cooperation between authorized programs running within a sysplex.

CTC. The channel-to-channel (CTC) channel can communicate with a CTC on another host for intersystem communication.

Customer Information Control System (CICS). A general-purpose transactional program that controls online communication between terminal users and a database for a large number of end users on a real-time basis.

customization dialogs. The customization dialogs are an ISPF application. They are used to customize the enterprise policy, like, for example, the enterprise resources and the relationships between resources, or the automation policy for systems in the enterprise. How to use these dialogs is described in *System Automation for z/OS Customizing and Programming*.

CVC. A channel operating in converted (CVC) mode transfers data in blocks and a CBY channel path transfers data in bytes. Converted CVC or CBY channel paths can communicate with a parallel control unit. This resembles a point-to-point parallel path and dedicated connection, regardless whether it passes through a switch.

D

DASD. Direct access storage device.

data services task (DST). The NetView subtask that gathers, records, and manages data in a VSAM file or a network device that contains network management information.

data set. The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access.

data set members. Members of partitioned data sets that are individually named elements of a larger file that can be retrieved by name.

DBCS. Double-byte character set.

DCCF. Disabled console communication facility.

DCF. Document composition facility.

DELAY Report. An RMF report that shows the activity of each job in the system and the hardware and software resources that are delaying each job.

Devices. You can see information about all devices (such as printers, tape or disk drives, displays, or communications controllers) attached to a particular switch, and control paths and jobs to devices.

DEVR Report. An RMF report that presents information about the activity of I/O devices that are delaying jobs.

dialog. Interactive 3270 panels.

direct access storage device (DASD). A device in which the access time is effectively independent of the location of the data; for example, a disk.

disabled console communication facility (DCCF). A z/OS component that provides limited-function console communication during system recovery situations.

display. (1) To present information for viewing, usually on the screen of a workstation or on a hardcopy device. (2) Deprecated term for *panel*.

disk operating system (DOS). (1) An operating system for computer systems that use disks and diskettes for auxiliary storage of programs and data. (2) Software for a personal computer that controls the processing of programs. For the IBM Personal Computer, the full name is Personal Computer Disk Operating System (PCDOS).

distribution manager. The component of the NetView program that enables the host system to use, send, and delete files and programs in a network of computers.

domain. (1) An access method and its application programs, communication controllers, connecting lines, modems, and attached workstations. (2) In SNA, a system services control point (SSCP) and the physical units (PUs), logical units (LUs), links, link stations, and associated resources that the SSCP can control by means of activation requests and deactivation requests.

double-byte character set (DBCS). A character set, such as Kanji, in which each character is represented by a 2-byte code.

DP enterprise. Data processing enterprise.

| **DSIPARM.** This file is a collection of members of
| NetView's customization.

DST. Data Services Task.

E

EBCDIC. Extended binary-coded decimal interchange code. A coded character set consisting of 8-bit coded characters.

ECB. Event control block. A control block used to represent the status of an event.

EMCS. Extended multiple console support.

enterprise. An organization, such as a business or a school, that uses data processing.

| **enterprise monitoring.** Enterprise monitoring is used
| by SA z/OS to update the *NetView Management Console*
| (*NMC*) resource status information that is stored in the
| *Resource Object Data Manager (RODM)*. Resource status
| information is acquired by enterprise monitoring of the
| *Resource Measurement Facility (RMF) Monitor III* service
| information at user-defined intervals. SA z/OS stores
| this information in its operational information base,
| where it is used to update the information presented to
| the operator in graphic displays.

entries. Resources, such as processors, entered on panels.

entry type. Resources, such as processors or applications, used for automation and monitoring.

environment. Data processing enterprise.

error threshold. An automation policy setting that specifies when SA z/OS should stop trying to restart or recover an application, subsystem or component, or offload a data set.

ESA. Enterprise Systems Architecture.

eServer. Processor family group designator used by the SA z/OS customization dialogs to define a target hardware as member of the zSeries or 390-CMOS processor families.

event. (1) In NetView, a record indicating irregularities of operation in physical elements of a network. (2) An occurrence of significance to a task; for example, the completion of an asynchronous operation, such as an input/output operation. (3) Events are part of a trigger condition, in a way that if all events of a trigger condition have occurred, a *STARTUP* or *SHUTDOWN* of an application is performed.

exception condition. An occurrence on a system that is a deviation from normal operation. SA z/OS monitoring highlights exception conditions and allows an SA z/OS enterprise to be managed by exception.

extended recovery facility (XRF). A facility that minimizes the effect of failures in z/OS, VTAM, the host processor, or high availability applications during sessions between high availability applications and designated terminals. This facility provides an alternate subsystem to take over sessions from the failing subsystem.

F

fallback system. See *secondary system*.

field. A collection of bytes within a record that are logically related and are processed as a unit.

file manager commands. A set of SA z/OS commands that read data from or write data to the automation control file or the operational information base. These commands are useful in the development of automation that uses SA z/OS facilities.

focal point. In NetView, the focal-point domain is the central host domain. It is the central control point for any management services element containing control of the network management data.

focus host. A processor with the role in the context of a unified system image

focal point system. (1) A system that can administer, manage, or control one or more target systems. There are a number of different focal point systems associated with IBM automation products. (2) **NMC focal point system.** The NMC focal point system is a NetView system with an attached workstation server and LAN that gathers information about the state of the network. This focal point system uses RODM to store the data it collects in the data model. The information stored in RODM can be accessed from any LAN-connected workstation with NetView Management Console installed. (3) **NPDA focal point system.** This is a NetView system that collects all the NPDA alerts that are generated within your enterprise. It is supported by NetView. If you have SA z/OS installed the NPDA focal point system must be the same as your NMC focal point system. The NPDA focal point system is also known as the *alert focal point system*. (4) **SA z/OS Processor Operations focal point system.** This is a NetView system that has SA z/OS host code installed. The SA z/OS Processor Operations focal point system receives messages from the systems and operator consoles of the machines that it controls. It provides full systems and operations console function for its target systems. It can be used to IPL these systems. Note that some restrictions apply to the Hardware Management Console for an S/390 microprocessor cluster. (5) **SA z/OS SDF focal point system.** The SA z/OS SDF focal point system is an SA z/OS NetView system that collects status information from other SA z/OS NetViews within your enterprise. (6) **Status focal point system.** In NetView, the system to which STATMON, VTAM and NLDM send status information on network resources. If you have a NMC focal point, it must be on the same system as the Status focal point. (7) **Hardware Management Console.** Although not listed as a focal point, the Hardware Management Console acts as a focal point for the console functions of an S/390 microprocessor cluster. Unlike all the other focal points in this definition, the

Hardware Management Console runs on a LAN-connected workstation,

frame. For a System/390 microprocessor cluster, a frame contains one or two central processor complexes (CPCs), support elements, and AC power distribution.

full-screen mode. In NetView, a form of panel presentation that makes it possible to display the contents of an entire workstation screen at once. Full-screen mode can be used for fill-in-the-blanks prompting. Contrast with *line mode*.

G

gateway session. An NetView-NetView Task session with another system in which the SA z/OS outbound gateway operator logs onto the other NetView session without human operator intervention. Each end of a gateway session has both an inbound and outbound gateway operator.

generic alert. Encoded alert information that uses code points (defined by IBM and possibly customized by users or application programs) stored at an alert receiver, such as NetView.

generic routines. In SA z/OS, a set of self-contained automation routines that can be called from the NetView automation table, or from user-written automation procedures.

group. A collection of target systems defined through configuration dialogs. An installation might set up a group to refer to a physical site or an organizational or application entity.

group entry. A construct, created with the customization dialogs, used to represent and contain policy for a group.

group entry type. A collection of target systems defined through the customization dialog. An installation might set up a group to refer to a physical site or an organizational entity. Groups can, for example, be of type STANDARD or SYSPLEX.

H

Hardware Management Console. A console used by the operator to monitor and control a System/390 microprocessor cluster.

Hardware Management Console Application (HWMCA). A direct-manipulation object-oriented graphical user interface that provides single point of control and single system image for hardware elements. HWMCA provides customer grouping support, aggregated and real-time system status using colors, consolidated hardware messages support, consolidated operating system messages support, consolidated

service support, and hardware commands targeted at a single system, multiple systems, or a customer group of systems.

heartbeat. In SA z/OS, a function that monitors the validity of the status forwarding path between remote systems and the NMC focal point, and monitors the availability of remote z/OS systems, to ensure that status information displayed on the SA z/OS workstation is current.

help panel. An online panel that tells you how to use a command or another aspect of a product.

hierarchy. In the NetView program, the resource types, display types, and data types that make up the organization, or levels, in a network.

high-level language (HLL). A programming language that does not reflect the structure of any particular computer or operating system. For the NetView program, the high-level languages are PL/I and C.

HLL. High-level language.

host system. In a coupled system or distributed system environment, the system on which the facilities for centralized automation run. SA z/OS publications refer to target systems or focal-point systems instead of hosts.

host (primary processor). The processor at which you enter a command (also known as the *issuing processor*).

HWMCA. Hardware Management Console Application. Application for the graphic hardware management console that monitors and controls a central processor complex. It is attached to a target processor (a system 390 microprocessor cluster) as a dedicated system console. This microprocessor uses OCF to process commands.

I

images. A grouping of processors and I/O devices that you define. You can define a single-image mode that allows a multiprocessor system to function as one central processor image.

IMS/VS. Information Management System/Virtual Storage.

inbound. In SA z/OS, messages sent to the focal-point system from the PC or target system.

inbound gateway operator. The automation operator that receives incoming messages, commands, and responses from the outbound gateway operator at the sending system. The inbound gateway operator handles communications with other systems using a gateway session.

Information Management System/Virtual Storage (IMS/VS). A database/data communication (DB/DC) system that can manage complex databases and networks. Synonymous with IMS.

INGEIO PROC. The I/O operations default procedure name; part of the SYS1.PROCLIB.

initial program load (IPL). (1) The initialization procedure that causes an operating system to commence operation. (2) The process by which a configuration image is loaded into storage at the beginning of a workday or after a system malfunction. (3) The process of loading system programs and preparing a system to run jobs.

initialize automation. SA z/OS-provided automation that issues the correct z/OS start command for each subsystem when SA z/OS is initialized. The automation ensures that subsystems are started in the order specified in the automation control file and that prerequisite applications are functional.

input/output support processor (IOSP). The hardware unit that provides I/O support functions for the primary support processor and maintenance support functions for the processor controller.

Interactive System Productivity Facility (ISPF). An IBM licensed program that serves as a full-screen editor and dialog manager. Used for writing application programs, it provides a means of generating standard screen panels and interactive dialogs between the application programmer and the terminal user.

interested operator list. The list of operators who are to receive messages from a specific target system.

internal token. A *logical token* (LTOK); name by which the I/O resource or object is known; stored in IODF.

IOCDs. I/O configuration data set. The data set that describes the I/O configuration.

I/O Ops. I/O operations.

IOSP. Input/Output Support Processor.

I/O operations. The part of SA z/OS that provides you with a single point of logical control for managing connectivity in your active I/O configurations. I/O operations takes an active role in detecting unusual conditions and lets you view and change paths between a processor and an I/O device, using dynamic switching (the ESCON director). Also known as I/O Ops.

I/O resource number. Combination of channel path identifier (CHPID), device number, etc. See internal token.

IPL. Initial program load.

ISA. Industry Standard Architecture.

ISPF. Interactive System Productivity Facility.

ISPF console. From this 3270-type console you are logged onto ISPF to use the runtime panels for I/O operations and SA z/OS customization panels.

issuing host. See *primary host*; the base program at which you enter a command for processing.

J

JCL. Job control language.

JES. Job entry subsystem.

job. (1) A set of data that completely defines a unit of work for a computer. A job usually includes all necessary computer programs, linkages, files, and instructions to the operating system. (2) An address space.

job control language (JCL). A problem-oriented language designed to express statements in a job that are used to identify the job or describe its requirements to an operating system.

job entry subsystem (JES). A facility for spooling, job queuing, and managing I/O. In SA z/OS publications, JES refers to JES2 or JES3, unless distinguished as being either one or the other.

K

Kanji. An ideographic character set used in Japanese. See also *double-byte character set*.

L

LAN. Local area network.

line mode. A form of screen presentation in which the information is presented a line at a time in the message area of the terminal screen. Contrast with *full-screen mode*.

link. (1) In SNA, the combination of the link connection and the link stations joining network nodes; for example, a System/370 channel and its associated protocols, a serial-by-bit connection under the control of synchronous data link control (SDLC). (2) In SA z/OS, link connection is the physical medium of transmission.

link-attached. Describes devices that are physically connected by a telecommunication line. Contrast with *channel-attached*.

Linux for zSeries and S/390. UNIX-like open source operating system conceived by Linus Torvalds and developed across the internet.

local. Pertaining to a device accessed directly without use of a telecommunication line. Synonymous with *channel-attached*.

local area network (LAN). (1) A network in which a set of devices is connected for communication. They can be connected to a larger network. See also *token ring*. (2) A network in which communications are limited to a moderately-sized geographic area such as a single office building, warehouse, or campus, and that do not generally extend across public rights-of-way.

logical partition (LP). A subset of the processor hardware that is defined to support an operating system. See also *logically partitioned (LPAR) mode*.

logical switch number (LSN). Assigned with the switch parameter of the CHPID macro of the IOCP.

logical token (LTOK). Resource number of an object in the IODF.

logical unit (LU). In SNA, a port through which an end user accesses the SNA network and the functions provided by system services control points (SSCPs). An LU can support at least two sessions — one with an SSCP and one with another LU — and may be capable of supporting many sessions with other LUs. See also *physical unit (PU)* and *system services control point (SSCP)*.

logical unit (LU) 6.2. A type of logical unit that supports general communications between programs in a distributed processing environment. LU 6.2 is characterized by (a) a peer relationship between session partners, (b) efficient use of a session for multiple transactions, (c) comprehensive end-to-end error processing, and (d) a generic application program interface (API) consisting of structured verbs that are mapped into a product implementation. Synonym for advanced program-to-program communications (APPC).

logically partitioned (LPAR) mode. A central processor mode that enables an operator to allocate system processor hardware resources among several logical partitions. Contrast with *basic mode*.

LOGR. The sysplex logger.

LP. Logical partition.

LPAR. Logically partitioned (mode).

LU. Logical unit.

LU-LU session. In SNA, a session between two logical units (LUs) in an SNA network. It provides communication between two end users, or between an end user and an LU services component.

LU 6.2. Logical unit 6.2.

LU 6.2 session. A session initiated by VTAM on behalf of an LU 6.2 application program, or a session initiated by a remote LU in which the application program specifies that VTAM is to control the session by using the APPCCMD macro.

M

| **MAT.** Deprecated term for NetView Automation Table.

MCA. Micro Channel* architecture.

MCS. Multiple console support.

member. A specific function (one or more modules/routines) of a multisystem application that is defined to XCF and assigned to a group by the multisystem application. A member resides on one system in the sysplex and can use XCF services to communicate (send and receive data) with other members of the same group.

| **message automation table (MAT).** Deprecated term
| for NetView Automation Table.

message class. A number that SA z/OS associates with a message to control routing of the message. During automated operations, the classes associated with each message issued by SA z/OS are compared to the classes assigned to each notification operator. Any operator with a class matching one of the message's classes receives the message.

message forwarding. The SA z/OS process of sending messages generated at an SA z/OS target system to the SA z/OS focal-point system.

message group. Several messages that are displayed together as a unit.

message monitor task. A task that starts and is associated with a number of communications tasks. Message monitor tasks receive inbound messages from a communications task, determine the originating target system, and route the messages to the appropriate target control tasks.

message processing facility (MPF). A z/OS table that screens all messages sent to the z/OS console. The MPF compares these messages with a customer-defined list of messages on which to automate, suppress from the z/OS console display, or both, and marks messages to automate or suppress. Messages are then broadcast on the subsystem interface (SSI).

message suppression. The ability to restrict the amount of message traffic displayed on the z/OS console.

Micro Channel architecture. The rules that define how subsystems and adapters use the Micro Channel bus in a computer. The architecture defines the services that each subsystem can or must provide.

microprocessor. A processor implemented on one or a small number of chips.

migration. Installation of a new version or release of a program to replace an earlier version or release.

MP. Multiprocessor.

MPF. Message processing facility.

| **MPFLSTSA.** The MPFLST member that is built by
| SA z/OS.

Multiple Virtual Storage (MVS). An IBM licensed program. MVS, which is the predecessor of OS/390, is an operating system that controls the running of programs on a System/390 or System/370 processor. MVS includes an appropriate level of the Data Facility Product (DFP) and Multiple Virtual Storage/Enterprise Systems Architecture System Product Version 5 (MVS/ESA SP5).

multiprocessor (MP). A CPC that can be physically partitioned to form two operating processor complexes.

multisystem application. An application program that has various functions distributed across z/OS images in a multisystem environment.

multisystem environment. An environment in which two or more z/OS images reside in one or more processors, and programs on one image can communicate with programs on the other images.

MVS. Multiple Virtual Storage, predecessor of z/OS.

MVS image. A single occurrence of the MVS/ESA operating system that has the ability to process work.

MVS/JES2. Multiple Virtual Storage/Job Entry System 2. A z/OS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In an installation with more than one processor, each JES2 processor independently controls its job input, scheduling, and output processing.

MVS/ESA. Multiple Virtual Storage/Enterprise Systems Architecture.

N

NAU. (1) Network accessible unit. (2) Network addressable unit.

NCCF. Network Communications Control Facility.

NCP. (1) Network Control Program (IBM licensed program). Its full name is Advanced Communications Function for the Network Control Program. Synonymous with *ACF/NCP*. (2) Network control program (general term).

NetView. An IBM licensed program used to monitor a network, manage it, and diagnose network problems. NetView consists of a command facility that includes a presentation service, command processors, automation based on command lists, and a transaction processing structure on which the session monitor, hardware monitor, and terminal access facility (TAF) network management applications are built.

network accessible unit (NAU). A logical unit (LU), physical unit (PU), control point (CP), or system services control point (SSCP). It is the origin or the destination of information transmitted by the path control network. Synonymous with *network addressable unit*.

network addressable unit (NAU). Synonym for *network accessible unit*.

NetView automation procedures. A sequence of commands, packaged as a NetView command list or a command processor written in a high-level language. An automation procedure performs automation functions and runs under the NetView program.

| **NetView automation table (AT).** A table against
| which the NetView program compares incoming
| messages. A match with an entry triggers the specified
| response. SA z/OS entries in the NetView automation
| table trigger an SA z/OS response to target system
| conditions. Formerly known as the message automation
| table (MAT).

NetView Command list language. An interpretive language unique to NetView that is used to write command lists.

NetView (NCCF) console. A 3270-type console for NetView commands and runtime panels for system operations and processor operations.

NetView Graphic Monitor Facility (NGMF).
Deprecated term for NetView Management Console.

NetView hardware monitor. The component of NetView that helps identify network problems, such as hardware, software, and microcode, from a central control point using interactive display techniques. Formerly called *network problem determination application*.

NetView log. The log in which NetView records events pertaining to NetView and SA z/OS activities.

NetView message table. See *NetView automation table*.

| **NetView Management Console (NMC).** A function of
| the NetView program that provides a graphic,
| topological presentation of a network that is controlled
| by the NetView program. It provides the operator
| different views of a network, multiple levels of
| graphical detail, and dynamic resource status of the
| network. This function consists of a series of graphic

| windows that allows you to manage the network
| interactively. Formerly known as the NetView Graphic
| Monitor Facility (NGMF).

NetView-NetView task (NNT). The task under which a cross-domain NetView operator session runs. Each NetView program must have a NetView-NetView task to establish one NNT session. See also *operator station task*.

NetView-NetView Task session. A session between two NetView programs that runs under a NetView-NetView Task. In SA z/OS, NetView-NetView Task sessions are used for communication between focal point and remote systems.

NetView paths via logical unit (LU 6.2). A type of network-accessible port (VTAM connection) that enables end users to gain access to SNA network resources and communicate with each other. LU 6.2 permits communication between processor operations and the workstation.

network. (1) An interconnected group of nodes. (2) In data processing, a user application network. See *SNA network*.

Network Communications Control Facility (NCCF). The operations control facility for the network. NCCF consists of a presentation service, command processors, automation based on command lists, and a transaction processing structure on which the network management applications NLDM and NPDA are built. NCCF is a precursor to the NetView command facility.

Network Control Program (NCP). An IBM licensed program that provides communication controller support for single-domain, multiple-domain, and interconnected network capability. Its full name is Advanced Communications Function for the Network Control Program.

Networking NetView. In SA z/OS the NetView that performs network management functions, such as managing the configuration of a network. In SA z/OS it is common to also route alerts to the Networking NetView.

Network Problem Determination Application (NPDA). An NCCF application that helps you identify network problems, such as hardware, software, and microcode, from a central control point using interactive display methods. The alert manager for the network. The precursor of the NetView hardware monitor.

NGMF. Deprecated term for NetView Management Console.

NGMF focal-point system. Deprecated term for NMC focal point system.

NIP. Nucleus initialization program.

NMC focal point system. See *focal point system*

NMC workstation. The NMC workstation is the primary way to dynamically monitor SA z/OS systems. From the windows, you see messages, monitor status, view trends, and react to changes before they cause problems for end users. You can use multiple windows to monitor multiple views of the system.

NNT. NetView-NetView task.

notification message. An SA z/OS message sent to a human notification operator to provide information about significant automation actions. Notification messages are defined using the customization dialogs.

notification operator. A NetView console operator who is authorized to receive SA z/OS notification messages. Authorization is made through the customization dialogs.

NPDA. Network Problem Determination Application.

NPDA focal-point system. See *focal-point system*.

NTRI. NCP/token-ring interconnection.

nucleus initialization program (NIP). The program that initializes the resident control program; it allows the operator to request last-minute changes to certain options specified during system generation.

O

objective value. An average Workflow or Using value that SA z/OS can calculate for applications from past service data. SA z/OS uses the objective value to calculate warning and alert thresholds when none are explicitly defined.

OCA. In SA z/OS, operator console A, the active operator console for a target system. Contrast with *OCB*.

OCB. In SA z/OS, operator console B, the backup operator console for a target system. Contrast with *OCA*.

OCF. Operations command facility.

OCF-based processor. A central processor complex that uses an operations command facility for interacting with human operators or external programs to perform operations management functions on the CPC.

OPC/A. Operations Planning and Control/Advanced.

OPC/ESA. Operations Planning and Control/Enterprise Systems Architecture.

operating system (OS). Software that controls the execution of programs and that may provide services such as resource allocation, scheduling, input/output

control, and data management. Although operating systems are predominantly software, partial hardware implementations are possible. (T)

operations. The real-time control of a hardware device or software function.

operations command facility (OCF). A facility of the central processor complex that accepts and processes operations management commands.

Operations Planning and Control/Advanced (OPC/A). A set of IBM licensed programs that automate, plan, and control batch workload. OPC/A analyzes system and workload status and submits jobs accordingly.

Operations Planning and Control/ESA (OPC/ESA). A set of IBM licensed programs that automate, plan, and control batch workload. OPC/ESA analyzes system and workload status and submits jobs accordingly. The successor to OPC/A.

operator. (1) A person who keeps a system running. (2) A person or program responsible for managing activities controlled by a given piece of software such as z/OS, the NetView program, or IMS. (3) A person who operates a device. (4) In a language statement, the lexical entity that indicates the action to be performed on operands.

operator console. (1) A functional unit containing devices that are used for communications between a computer operator and a computer. (T) (2) A display console used for communication between the operator and the system, used primarily to specify information concerning application programs and I/O operations and to monitor system operation. (3) In SA z/OS, a console that displays output from and sends input to the operating system (z/OS, LINUX, VM, VSE). Also called *operating system console*. In the SA z/OS operator commands and configuration dialogs, OC is used to designate a target system operator console.

operator station task (OST). The NetView task that establishes and maintains the online session with the network operator. There is one operator station task for each network operator who logs on to the NetView program.

operator view. A set of group, system, and resource definitions that are associated together for monitoring purposes. An operator view appears as a graphic display in the graphic interface showing the status of the defined groups, systems, and resources.

OperatorView entry. A construct, created with the customization dialogs, used to represent and contain policy for an operator view.

OS. Operating system.

z/OS component. A part of z/OS that performs a specific z/OS function. In SA z/OS, component refers to entities that are managed by SA z/OS automation.

z/OS subsystem. Software products that augment the z/OS operating system. JES and TSO/E are examples of z/OS subsystems. SA z/OS includes automation for some z/OS subsystems.

z/OS system. A z/OS image together with its associated hardware, which collectively are often referred to simply as a system, or z/OS system.

OSA. I/O operations can display the open system adapter (OSA) channel logical definition, physical attachment, and status. You can configure an OSA channel on or off.

OST. Operator station task.

outbound. In SA z/OS, messages or commands from the focal-point system to the target system.

outbound gateway operator. The automation operator that establishes connections to other systems. The outbound gateway operator handles communications with other systems through a gateway session. The automation operator sends messages, commands, and responses to the inbound gateway operator at the receiving system.

P

page. (1) The portion of a panel that is shown on a display surface at one time. (2) To transfer instructions, data, or both between real storage and external page or auxiliary storage.

panel. (1) A formatted display of information that appears on a terminal screen. Panels are full-screen 3270-type displays with a monospaced font, limited color and graphics. (2) By using SA z/OS panels you can see status, type commands on a command line using a keyboard, configure your system, and passthru to other consoles. See also *help panel*. (3) In computer graphics, a display image that defines the locations and characteristics of display fields on a display surface. Contrast with *screen*.

parallel channels. Parallel channels operate in either byte (BY) or block (BL) mode. You can change connectivity to a parallel channel operating in block mode.

parameter. (1) A variable that is given a constant value for a specified application and that may denote the application. (2) An item in a menu for which the user specifies a value or for which the system provides a value when the menu is interpreted. (3) Data passed to a program or procedure by a user or another program, namely as an operand in a language statement, as an item in a menu, or as a shared data structure.

partition. (1) A fixed-size division of storage. (2) In VSE, a division of the virtual address area that is available for program processing. (3) On an IBM Personal Computer fixed disk, one of four possible storage areas of variable size; one can be accessed by DOS, and each of the others may be assigned to another operating system.

partitionable CPC. A CPC that can be divided into 2 independent CPCs. See also *physical partition*, *single-image mode*, *MP*, *side*.

partitioned data set (PDS). A data set in direct access storage that is divided into partitions, called *members*, each of which can contain a program, part of a program, or data.

passive monitoring. In SA z/OS, the receiving of unsolicited messages from z/OS systems and their resources. These messages can prompt updates to resource status displays. See also *active monitoring*.

PCE. Processor controller. Also known as the “support processor” or “service processor” in some processor families.

| **PDB.** Policy Database

PDS. Partitioned data set.

physical partition. Part of a CPC that operates as a CPC in its own right, with its own copy of the operating system.

physical unit (PU). In SNA, the component that manages and monitors the resources (such as attached links and adjacent link stations) of a node, as requested by a system services control point (SSCP) through an SSCP-PU session. An SSCP activates a session with the physical unit to indirectly manage, through the PU, resources of the node such as attached links.

physically partitioned (PP) configuration. A mode of operation that allows a multiprocessor (MP) system to function as two or more independent CPCs having separate power, water, and maintenance boundaries. Contrast with *single-image (SI) configuration*.

POI. Program operator interface.

policy. The automation and monitoring specifications for an SA z/OS enterprise. See *System Automation for z/OS Defining Automation Policy*.

| **policy database.** The database where the automation policy is recorded. Also known as the PDB.

POR. Power-on reset.

port. (1) System hardware to which the I/O devices are attached. (2) On an ESCON switch, a port is an addressable connection. The switch routes data through the ports to the channel or control unit. Each port has a name that can be entered into a switch matrix, and you

can use commands to change the switch configuration. (3) An access point (for example, a logical unit) for data entry or exit. (4) A functional unit of a node through which data can enter or leave a data network. (5) In data communication, that part of a data processor that is dedicated to a single data channel for the purpose of receiving data from or transmitting data to one or more external, remote devices. (6) power-on reset (POR) (7) A function that re-initializes all the hardware in a CPC and loads the internal code that enables the CPC to load and run an operating system.

PP. Physically partitioned (configuration).

PPT. Primary POI task.

primary host. The base program at which you enter a command for processing.

primary POI task (PPT). The NetView subtask that processes all unsolicited messages received from the VTAM program operator interface (POI) and delivers them to the controlling operator or to the command processor. The PPT also processes the initial command specified to execute when NetView is initialized and timer request commands scheduled to execute under the PPT.

primary system. A system is a primary system for an application if the application is normally meant to be running there. SA z/OS starts the application on all the primary systems defined for it.

problem determination. The process of determining the source of a problem; for example, a program component, machine failure, telecommunication facilities, user or contractor-installed programs or equipment, environment failure such as a power loss, or user error.

processor controller. Hardware that provides support and diagnostic functions for the central processors.

processor operations. The part of SA z/OS that monitors and controls processor (hardware) operations. Processor operations provides a connection from a focal-point system to a target system. Through NetView on the focal-point system, processor operations automates operator and system consoles for monitoring and recovering target systems. Also known as ProcOps.

processor operations control file. Named by your system programmer, this file contains configuration and customization information. The programmer records the name of this control file in the processor operations file generation panel ISQDPG01.

Processor Resource/Systems Manager (PR/SM). The feature that allows the processor to use several operating system images simultaneously and provides logical partitioning capability. See also *LPAR*.

ProcOps. Processor operations.

| **ProcOps Service Machine (PSM).** The PSM is a CMS user on a VM host system. It runs a CMS multitasking application that serves as "virtual hardware" for ProcOps. ProcOps communicates via the PSM with the VM guest systems that are defined as target systems within ProcOps.

product automation. Automation integrated into the base of SA z/OS for the products DB2, CICS, IMS, OPC (formerly called *features*).

program to program interface (PPI). A NetView function that allows user programs to send or receive data buffers from other user programs and to send alerts to the NetView hardware monitor from system and application programs.

protocol. In SNA, the meanings of, and the sequencing rules for, requests and responses used for managing the network, transferring data, and synchronizing the states of network components.

proxy resource. A resource defined like an entry type APL representing a processor operations target system.

PR/SM. Processor Resource/Systems Manager.

| **PSM.** ProcOps Service Machine.

PU. Physical unit.

R

remote system. A system that receives resource status information from an SA z/OS focal-point system. An SA z/OS remote system is defined as part of the same SA z/OS enterprise as the SA z/OS focal-point system to which it is related.

requester. A requester is a workstation software, which enables users to log on to a domain, that is, to the server(s) belonging to this domain, and use the resources in this domain. After the log on to a domain, users can access the shared resources and use the processing capability of the server(s). Because the bigger part of shared resources is on the server(s), users can reduce hardware investment.

resource. (1) Any facility of the computing system or operating system required by a job or task, and including main storage, input/output devices, the processing unit, data sets, and control or processing programs. (2) In NetView, any hardware or software that provides function to the network. (3) In SA z/OS, any z/OS application, z/OS component, job, device, or target system capable of being monitored or automated through SA z/OS.

Resource Access Control Facility (RACF). A program that can provide data security for all your resources. RACF protects data from accidental or deliberate unauthorized disclosure, modification, or destruction.

resource group. A physically partitionable portion of a processor. Also known as a *side*.

Resource Monitoring Facility (RMF) Monitor III. A program that measures and reports on the availability and activity of system hardware and software resources, such as processors, devices, storage, and address spaces. RMF can issue online reports about system performance problems as they occur.

Resource Object Data Manager (RODM). A data cache manager designed to support process control and automation applications. RODM provides an in-memory data cache for maintaining real-time data in an address space that is accessible by multiple applications. RODM also allows an application to query an object and receive a rapid response and act on it.

resource token. A unique internal identifier of an ESCON resource or resource number of the object in the IODF.

restart automation. SA z/OS-provided automation that monitors subsystems to ensure that they are running. If a subsystem fails, SA z/OS attempts to restart it according to the policy in the automation control file.

Restructured Extended Executor (REXX). An interpretive language used to write command lists.

return code. A code returned from a program used to influence the issuing of subsequent instructions.

REXX. Restructured Extended Executor.

REXX procedure. A command list written with the Restructured Extended Executor (REXX), which is an interpretive language.

RMF. Resource Measurement Facility.

RODM. Resource Object Data Manager.

S

SAF. Security Authorization Facility.

SA z/OS. System Automation for z/OS

SA z/OS customization dialogs. An ISPF application through which the SA z/OS policy administrator defines policy for individual z/OS systems and builds automation control data and RODM load function files.

SA z/OS customization focal point system. See *focal point system*.

SA z/OS data model. The set of objects, classes and entity relationships necessary to support the function of SA z/OS and the NetView automation platform.

SA z/OS enterprise. The group of systems and resources defined in the customization dialogs under one enterprise name. An SA z/OS enterprise consists of connected z/OS systems running SA z/OS.

SA z/OS focal point system. See *focal point system*.

SA z/OS policy. The description of the systems and resources that make up an SA z/OS enterprise, together with their monitoring and automation definitions.

SA z/OS policy administrator. The member of the operations staff who is responsible for defining SA z/OS policy.

SA z/OS satellite. If you are running two NetViews on an z/OS system to split the automation and networking functions of NetView, it is common to route alerts to the Networking NetView. For SA z/OS to process alerts properly on the Networking NetView, you must install a subset of SA z/OS code, called an *SA z/OS satellite* on the Networking NetView.

| **SA z/OS SDF focal point system.** See *focal point system*.

SCA. In SA z/OS, system console A, the active system console for a target hardware. Contrast with *SCB*.

SCB. In SA z/OS, system console B, the backup system console for a target hardware. Contrast with *SCA*.

screen. Deprecated term for display panel.

screen handler. In SA z/OS, software that interprets all data to and from a full-screen image of a target system. The interpretation depends on the format of the data on the full-screen image. Every processor and operating system has its own format for the full-screen image. A screen handler controls one PS/2 connection to a target system.

SDF. Status Display Facility.

SDLC. Synchronous data link control.

SDSF. System Display and Search Facility.

secondary system. A system is a secondary system for an application if it is defined to automation on that system, but the application is not normally meant to be running there. Secondary systems are systems to which an application can be moved in the event that one or more of its primary systems are unavailable. SA z/OS does not start the application on its secondary systems.

server. A server is a workstation that shares resources, which include directories, printers, serial devices, and computing powers.

service language command (SLC). The line-oriented command language of processor controllers or service processors.

service processor (SVP). The name given to a processor controller on smaller System/370 processors.

service period. Service periods allow the users to schedule the availability of applications. A service period is a set of time intervals (service windows), during which an application should be active.

service threshold. An SA z/OS policy setting that determines when to notify the operator of deteriorating service for a resource. See also *alert threshold* and *warning threshold*.

session. In SNA, a logical connection between two network addressable units (NAUs) that can be activated, tailored to provide various protocols, and deactivated, as requested. Each session is uniquely identified in a transmission header by a pair of network addresses identifying the origin and destination NAUs of any transmissions exchanged during the session.

session monitor. The component of the NetView program that collects and correlates session-related data and provides online access to this information. The successor to NLDM.

shutdown automation. SA z/OS-provided automation that manages the shutdown process for subsystems by issuing shutdown commands and responding to prompts for additional information.

side. A part of a partitionable CPC that can run as a physical partition and is typically referred to as the A-side or the B-side.

Simple Network Management Protocol (SNMP). An IP based industry standard protocol to monitor and control resources in an IP network.

single image. A processor system capable of being physically partitioned that has not been physically partitioned. Single-image systems can be target hardware processors.

single-image (SI) mode. A mode of operation for a multiprocessor (MP) system that allows it to function as one CPC. By definition, a uniprocessor (UP) operates in single-image mode. Contrast with *physically partitioned (PP) configuration*.

SLC. Service language command.

SMP/E. System Modification Program Extended.

SNA. Systems Network Architecture.

SNA network. In SNA, the part of a user-application network that conforms to the formats and protocols of systems network architecture. It enables reliable

transfer of data among end users and provides protocols for controlling the resources of various network configurations. The SNA network consists of network addressable units (NAUs), boundary function components, and the path control network.

SNMP. Simple Network Management Protocol (a TCP/IP protocol). A protocol that allows network management by elements, such as gateways, routers, and hosts. This protocol provides a means of communication between network elements regarding network resources.

solicited message. An SA z/OS message that directly responds to a command. Contrast with *unsolicited message*.

SSCP. System services control point.

SSI. Subsystem interface.

start automation. SA z/OS-provided automation that manages and completes the startup process for subsystems. During this process, SA z/OS replies to prompts for additional information, ensures that the startup process completes within specified time limits, notifies the operator of problems, if necessary, and brings subsystems to an UP (or ready) state.

startup. The point in time at which a subsystem or application is started.

status. The measure of the condition or availability of the resource.

status focal-point system. See *focal—point system*.

status display facility (SDF). The system operations part of SA z/OS that displays status of resources such as applications, gateways, and write-to-operator messages (WTORs) on dynamic color-coded panels. SDF shows spool usage problems and resource data from multiple systems.

steady state automation. The routine monitoring, both for presence and performance, of subsystems, applications, volumes and systems. Steady state automation may respond to messages, performance exceptions and discrepancies between its model of the system and reality.

structure. A construct used by z/OS to map and manage storage on a coupling facility. See cache structure, list structure, and lock structure.

subgroup. A named set of systems. A subgroup is part of an SA z/OS enterprise definition and is used for monitoring purposes.

SubGroup entry. A construct, created with the customization dialogs, used to represent and contain policy for a subgroup.

subsystem. (1) A secondary or subordinate system, usually capable of operating independent of, or asynchronously with, a controlling system. (2) In SA z/OS, an z/OS application or subsystem defined to SA z/OS.

subsystem interface. The z/OS interface over which all messages sent to the z/OS console are broadcast.

support element. A hardware unit that provides communications, monitoring, and diagnostic functions to a central processor complex (CPC).

support processor. Another name given to a processor controller on smaller System/370 processors; see *service processor*.

SVP. Service processor.

switches. ESCON directors are electronic units with ports that dynamically switch to route data to I/O devices. The switches are controlled by I/O operations commands that you enter on a workstation.

switch identifier. The switch device number (swchdevn), the logical switch number (LSN) and the switch name

symbolic destination name (SDN). Used locally at the workstation to relate to the VTAM application name.

synchronous data link control (SDLC). A discipline for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges may be duplex or half-duplex over switched or nonswitched links. The configuration of the link connection may be point-to-point, multipoint, or loop. SDLC conforms to subsets of the Advanced Data Communication Control Procedures (ADCCP) of the American National Standards Institute and High-Level Data Link Control (HDLC) of the International Standards Organization.

SYSINFO Report. An RMF report that presents an overview of the system, its workload, and the total number of jobs using resources or delayed for resources.

SysOps. System operations.

sysplex. A set of z/OS systems communicating and cooperating with each other through certain multisystem hardware components (coupling devices and timers) and software services (couple data sets).

In a sysplex, z/OS provides the coupling services that handle the messages, data, and status for the parts of a multisystem application that has its workload spread across two or more of the connected processors, sysplex timers, coupling facilities, and couple data sets (which contains policy and states for automation).

A parallel sysplex is a sysplex that includes a coupling facility.

sysplex application group. A sysplex application group is a grouping of applications that can run on any system in a sysplex.

sysplex couple data set. A couple data set that contains sysplex-wide data about systems, groups, and members that use XCF services. All z/OS systems in a sysplex must have connectivity to the sysplex couple data set. See also *couple data set*.

Sysplex Timer. An IBM unit that synchronizes the time-of-day (TOD) clocks in multiple processors or processor sides. External Time Reference (ETR) is the z/OS generic name for the IBM Sysplex Timer (9037).

system. In SA z/OS, system means a focal point system (z/OS) or a target system (MVS, VM, VSE, LINUX, or CF).

| **System Automation for z/OS.** The full name for
| SA z/OS.

| **System Automation for OS/390.** The full name for
| SA OS/390, the predecessor to System Automation for
| z/OS.

system console. (1) A console, usually having a keyboard and a display screen, that is used by an operator to control and communicate with a system. (2) A logical device used for the operation and control of hardware functions (for example, IPL, alter/display, and reconfiguration). The system console can be assigned to any of the physical displays attached to a processor controller or support processor. (3) In SA z/OS, the hardware system console for processor controllers or service processors of processors connected using SA z/OS. In the SA z/OS operator commands and configuration dialogs, SC is used to designate the system console for a target hardware processor.

System Display and Search Facility (SDSF). An IBM licensed program that provides information about jobs, queues, and printers running under JES2 on a series of panels. Under SA z/OS you can select SDSF from a pull-down menu to see the resources' status, view the z/OS system log, see WTOR messages, and see active jobs on the system.

System entry. A construct, created with the customization dialogs, used to represent and contain policy for a system.

System Modification Program/Extended (SMP/E). An IBM licensed program that facilitates the process of installing and servicing an z/OS system.

system operations. The part of SA z/OS that monitors and controls system operations applications and subsystems such as NetView, SDSF, JES, RMF, TSO, RODM, ACF/VTAM, CICS, IMS, and OPC. Also known as SysOps.

system services control point (SSCP). In SNA, the focal point within an SNA network for managing the configuration, coordinating network operator and problem determination requests, and providing directory support and other session services for end users of the network. Multiple SSCPs, cooperating as peers, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its domain.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks.

System/390 microprocessor cluster. A configuration that consists of central processor complexes (CPCs) and may have one or more integrated coupling facilities.

T

TAF. Terminal access facility.

target. A processor or system monitored and controlled by a focal-point system.

target control task. In SA z/OS, target control tasks process commands and send data to target systems and workstations through communications tasks. A target control task (a NetView autotask) is assigned to a target system when the target system is initialized.

target hardware. In SA z/OS, the physical hardware on which a target system runs. It can be a single-image or physically partitioned processor. Contrast with *target system*.

target system. (1) In a distributed system environment, a system that is monitored and controlled by the focal-point system. Multiple target systems can be controlled by a single focal-point system. (2) In SA z/OS, a computer system attached to the focal-point system for monitoring and control. The definition of a target system includes how remote sessions are established, what hardware is used, and what operating system is used.

task. (1) A basic unit of work to be accomplished by a computer. (2) In the NetView environment, an operator station task (logged-on operator), automation operator (autotask), application task, or user task. A NetView task performs work in the NetView environment. All SA z/OS tasks are NetView tasks. See also *communications task*, *message monitor task*, and *target control task*.

telecommunication line. Any physical medium, such as a wire or microwave beam, that is used to transmit data.

terminal access facility (TAF). (1) A NetView function that allows you to log onto multiple applications either on your system or other systems. You can define TAF sessions in the SA z/OS customization panels so you don't have to set them up each time you want to use them. (2) In NetView, a facility that allows a network operator to control a number of subsystems. In a full-screen or operator control session, operators can control any combination of subsystems simultaneously.

terminal emulation. The capability of a microcomputer or personal computer to operate as if it were a particular type of terminal linked to a processing unit to access data.

threshold. A value that determines the point at which SA z/OS automation performs a predefined action. See *alert threshold*, *warning threshold*, and *error threshold*.

time of day (TOD). Typically refers to the time-of-day clock.

Time Sharing Option (TSO). An optional configuration of the operating system that provides conversational time sharing from remote stations. It is an interactive service on z/OS, MVS/ESA, and MVS/XA.

Time-Sharing Option/Extended (TSO/E). An option of z/OS that provides conversational timesharing from remote terminals. TSO/E allows a wide variety of users to perform many different kinds of tasks. It can handle short-running applications that use fewer sources as well as long-running applications that require large amounts of resources.

timers. A NetView command that issues a command or command processor (list of commands) at a specified time or time interval.

TOD. Time of day.

token ring. A network with a ring topology that passes tokens from one attaching device to another; for example, the IBM Token-Ring Network product.

TP. Transaction program.

transaction program. In the VTAM program, a program that performs services related to the processing of a transaction. One or more transaction programs may operate within a VTAM application program that is using the VTAM application program interface (API). In that situation, the transaction program would request services from the applications program using protocols defined by that application program. The application program, in turn, could request services from the VTAM program by issuing the APPCCMD macro instruction.

transitional automation. The actions involved in starting and stopping subsystems and applications that

have been defined to SA z/OS. This can include issuing commands and responding to messages.

translating host. Role played by a host that turns a resource number into a token during a unification process.

trigger. Triggers, in combination with events and service periods, are used to control the starting and stopping of applications in a single system or a parallel sysplex.

TSO. Time Sharing Option.

TSO console. From this 3270-type console you are logged onto TSO or ISPF to use the runtime panels for I/O operations and SA z/OS customization panels.

TSO/E. TSO Extensions.

U

UCB. The unit control block; an MVS/ESA data area that represents a device and that is used for allocating devices and controlling I/O operations.

unsolicited message. An SA z/OS message that is not a direct response to a command. Contrast with *solicited message*.

user task. An application of the NetView program defined in a NetView TASK definition statement.

Using. An RMF Monitor III definition. Jobs getting service from hardware resources (processors or devices) are **using** these resources. The use of a resource by an address space can vary from 0% to 100% where 0% indicates no use during a Range period, and 100% indicates that the address space was found using the resource in every sample during that period. See also *Workflow*.

V

view. In the NetView Graphic Monitor Facility, a graphical picture of a network or part of a network. A view consists of nodes connected by links and may also include text and background lines. A view can be displayed, edited, and monitored for status information about network resources.

Virtual Storage Extended (VSE). An IBM licensed program whose full name is Virtual Storage Extended/Advanced Function. It is an operating system that controls the execution of programs.

Virtual Telecommunications Access Method (VTAM). An IBM licensed program that controls communication and the flow of data in an SNA network. It provides single-domain, multiple-domain, and interconnected network capability. Its full name is Advanced

Communications Function for the Virtual Telecommunications Access Method. Synonymous with *ACF/VTAM*.

VM/ESA. Virtual Machine/Enterprise Systems Architecture.

| **VM Second Level Systems Support.** With this
| function, Processor Operations is able to control VM
| second level systems (VM guest systems) in the same
| way that it controls systems running on real hardware.

volume. A direct access storage device (DASD) volume or a tape volume that serves a system in an SA z/OS enterprise.

volume entry. A construct, created with the customization dialogs, used to represent and contain policy for a volume.

volume group. A named set of volumes. A volume group is part of a system definition and is used for monitoring purposes.

volume group entry. A construct, created with the customization dialogs, used to represent and contain policy for a volume group.

Volume Workflow. The SA z/OS Volume Workflow variable is derived from the RMF Resource Workflow definition, and is used to measure the performance of volumes. SA z/OS calculates Volume Workflow using:

$$\text{Volume Workflow \%} = \frac{\text{accumulated Using}}{\text{accumulated Using} + \text{accumulated Delay}} * 100$$

The definition of **Using** is the percentage of time when a job has had a request accepted by a channel for the volume, but the request is not yet complete.

The definition of **Delay** is the delay that waiting jobs experience because of contention for the volume. See also *Address Space Workflow*.

VSE. Virtual Storage Extended.

VTAM. Virtual Telecommunications Access Method.

W

warning threshold. An application or volume service value that determines the level at which SA z/OS changes the associated icon in the graphic interface to the warning color. See *alert threshold*.

workflow. See *Address Space Workflow* and *Volume Workflow*.

workstation. In SA z/OS workstation means the *graphic workstation* that an operator uses for day-to-day operations.

write-to-operator (WTO). A request to send a message to an operator at the z/OS operator console. This request is made by an application and is handled by the WTO processor, which is part of the z/OS supervisor program.

write-to-operator-with-reply (WTOR). A request to send a message to an operator at the z/OS operator console that requires a response from the operator. This request is made by an application and is handled by the WTO processor, which is part of the z/OS supervisor program.

WTO. Write-to-Operator.

WTOR. Write-to-Operator-with-Reply.

WWV. The US National Institute of Standards and Technology (NIST) radio station that provides standard time information. A second station, known as WWVB, provides standard time information at a different frequency.

X

XCF. Cross-system coupling facility.

XCF couple data set. The name for the sysplex couple data set prior to MVS/ESA System Product Version 5 Release 1. See also *sysplex couple data set*.

XCF group. A set of related members that a multisystem application defines to XCF. A member is a specific function, or instance, of the application. A member resides on one system and can communicate with other members of the same group across the sysplex.

XRF. Extended recovery facility.

Numerics

390-CMOS. Processor family group designator used in the SA z/OS processor operations documentation and in the online help to identify any of the following S/390 CMOS processor machine types: 9672, 9674, 2003, 3000, or 7060. SA z/OS processor operations uses the OCF facility of these processors to perform operations management functions. See *OCF-based processor*.

Index

A

ABENDING automation status 38
ACF 11
ACF REFRESH 61
ACTIVE automation status 39
additional BCP Internal Interfaces 281
advanced automation options
 application monitor 49
alert 241, 242
alert adapter 249
alerts 242
 monitoring 133
alternate CDS
 allocating 160
 spare volumes 160
 switch to primary CDS 160
 turn into primary CDS 160
AMC file 61
anchors 220, 221
 lost&found 222
AOCHHELP command 69
AOCTRACE command 276, 278
application group 12, 107
application monitor status 45
application move 22, 109
applications 12
 starting and stopping 58
ASSIST automation status 39
AUTODOWN automation status 39
Automatic Restart Manager
 statuses 46
Automatic Restart Manager status 46
automation agent 13, 55
automation agent flags 31, 103
automation agent status 38
 changing 114
automation agent view with
 DISPSTAT 77
automation control file 8, 11
 identifying 62
 reloading 61
Automation flag 31
automation flags 31
 Automation 31
 automation agent flags 31
 determining which are on 103
 displaying 105
 global automation flag 32
 Initstart 31
 override settings 105
 Recover 31
 Restart 32
 Shutdown 32
automation initialization options 259
automation manager 13
 diagnostic information 55
 making it primary 54
 operation mode 113
 retrieve information from 17
 starting 53
 stopping 54

automation manager (*continued*)
 stopping with different options 54
automation manager and automation
 agents cooperation 21
automation manager configuration file 8
 identifying 63
automation manager interaction 113
automation manager messages
 log 276
automation manager status 13
automation operator 11
automation policy 8, 11
automation status 33, 35
automation table 62
 replacing or reloading 62
autotasks 11
AUTOTBL command 62
AUTOTERM automation status 40

B

backup
 disruptive 27
 non-disruptive 26
base preference 24
BASIC 108
basic group start type 51
basic group stop type 51
basic mode 7
BCP Internal Interface
 communication task names,
 determining 281
 First Failure Data Capture
 support 279
 recovery actions 281
 terminating after recovery
 actions 283
 tracing and debugging 278
 tracing specific connections 279
 using AOCTRACE 278
BCP Internal Interfaces
 additional 281
best practices
 HealthChecker 172
BREAKING automation status 40
BROKEN automation status 40

C

central processor 6
CFRM policy
 pending changes 157
 switching 162
changing automation agent status 114
changing policy for resources 109
channel
 types 9
channel paths 202
CLIST 11

clocks
 time of day 5
colors
 gateway sessions 136
 in DISPSTAT 135
 in SDF 135
 WTOR 136
command
 dialogs 68
 issuing 67
Command Dialogs Main Menu 68
command list (CLIST) 11
command processor 11
Command Profile Editor 67
commands
 AOCHHELP 69
 AOCTRACE 276
 DISPGW 184
 DISPMTR 125
 DISPSTAT 77
 DISPWTOR 117
 EXPLAIN 69
 HELPMMSG 69
 INGHC 175
 INGHC, check filters 177
 INGHC, results 177
 INGHC, using 176
 INGLKUP 183
 INGNTFY 121
 INGPLEX BESTpractices 172
 INGPLEX SDUMP 166
 INGPLEX SLIP 171
 INGPLEX SVCDUMP 168
 INGRPT 85
 INGSEND 184
 INGTHRES 180
 ISQCMON 118
 ISQHELP 69
 ISQTRACE 212
 ISQVARS 127, 129, 130
 ISQXDST 118, 119, 127
 ISQXMON 118
 issuing via NetView Management
 Console 217
 RMTCMD 184
 ROUTE 184
 SDF 133, 140, 141
 SETSTATE 114, 115
 SETTIMER 112
commands from NMC 226
communicating
 with other systems 184
communications path 7
 monitoring status 131
communications task 7
compound status 33, 37
concurrent user access 273
conflicting requests 20
connections, tracing specific 279
connectivity changes 202
control units 8

- cooperation of automation manager and
 - automation agents 21
- correlation rules 249
- couple data set (CDS)
 - alternate CDS
 - allocating 160
 - spare volumes 160
 - turn into primary CDS 160
- couple data sets 158
- coupling facilities 8
 - channels 10
- CPE 67
- critical threshold 179
- cross-domain session 186
- CTLDOWN automation status 40

D

- DASD 8
- DB2
 - precheck option 93
- debugging 274
 - BCP Internal Interface 278
 - First Failure Data Capture support 279
 - ProcOps 278
- dependency information 84
- dependency relationships 20
- desired status 33, 36
- determining
 - BCP Internal Interface communication task names 281
- devices 8, 202
- DIAGINFO 55
- diagnostic functions 277
- diagnostic information for the automation manager 55
- dialog
 - ProcOps 207
- dialog help 69
- DISPASF 180, 182
- DISPERRS 180, 181
- DISPEVT 101
- DISPEVTS 101, 102
- DISPFLGS 105
- DISPINFO 81
- displaying
 - automation flags 105
- displaying statistical information 85
- displaying status
 - DISPSTAT 77
- DISPMTR
 - command 125
 - panel 125
- DISPNTFY (Display Notification Operator Settings) 120
- DISPSCHD command dialog 105
- DISPSFLT 79
- DISPSTAT 77, 103
- DISPSTAT command dialog 103
- DISPTREE 84
- DISPTRG 99, 198
- DISPWTOR (Display WTORs) 117
- DISPWTOR command 117
- disruptive restoration 27
- documents, licensed xvi
- DOWN automation status 41

- duplexing
 - system-managed 158
 - user-managed 158
- dynamic switches 8
- dynamic takeover file 56

E

- ENDED automation status 41
- ENDING automation status 41
- enterprise systems connection (ESCON) 8
- error thresholds 179
 - set or change 183
- errors
 - display occurred 181, 182
- ESCON
 - converter 8
 - director 8
 - switch 8
- event information
 - DISPEVT 101
 - DISPEVTS 101
 - INGEVENT 101
- Event/Automation Service 241
- events (NMC) 221
- exception views 230, 235
- EXPLAIN command 69
- EXTSTART automation status 41

F

- FALLBACK 135
- FALLBACK automation status 42
- filters
 - DISPSFLT 79
 - INGFILT 73
- focal point monitoring 134
- focal point processor 6

G

- gateway display 190
- gateway sessions
 - colors 136
 - determining status of 190
 - monitoring with SDF 133
 - overview 190
- GDG 62
- global automation flag 32
- global resource serialization 273
- goal 18
- goal driven automation 14
- goals
 - creation of 14
- group start and stop types, using 51
- grouped resource objects 12
- GRS 273
- guest machines
 - processor operations support 207
- guest systems hierarchy 207
- guests summary panel ISQETRGG 210

H

- HALFDOWN automation status 42
- HALTED automation status 42
- Handling Jobs that are not Controlled by SA z/OS 183
- hardware configuration overview 5
- hardware monitoring 122
- health performance
 - monitoring 124
- health status 33, 37
- HealthChecker
 - best practices 172
 - check filters 177
 - reports 175
 - reports, about 175
 - reports, recommended procedure 176
 - reports, types of actions 175
 - results 177
- heartbeat 222
- heartbeat function 225
- help 69
 - automation manager status 69
 - automation status 69
 - dialog help 69
 - EXPLAIN command 69
 - HELPMSG command 69
 - ISQHELP command 69
 - message help 69
- HELPMSG command 69
- hierarchy
 - guest systems 207
- hold flag 33
- HSACTWR 274
- HSAJPTRC 276

I

- I/O operations 5
 - connectivity changes 202
 - introduction 201
 - starting and stopping 57
- I/O operations commands
 - using the z/OS operator console 204
- I/O operations ISPF dialog 204, 263
 - help 271
 - tutorial 271
- I/O operations ISPF Main Menu 264
- I/O resources 202
- image 7
- INACTIVE automation status 42
- INGAMS 113
 - diagnostic option 277
- INGAMS REFRESH 61
- INGAUTO 106
- INGDLG 263
- INGEIO 58
- INGEVENT 94, 101
- INGFILT 73
- INGGROUP 108
- INGHC 175
- INGHC command 175
 - check filters 177
 - results 177
 - using 176
- INGHIST 96
 - display a log 276

INGINFO 74, 193
 INGLIST 71, 191, 193
 INGLKUP 183
 INGNIFY 120, 121
 INGNIFY command 121
 INGNIFY command dialog 121
 INGPDB 273
 INGPLEX 141
 INGPLEX CDS 158
 INGPLEX CF 146
 INGPLEX command
 BESTpractices 172
 SDUMP 166
 SLIP 171
 SVCDUMP 168
 INGPLEX CONsole 144
 INGPLEX DUMP 166
 INGPLEX IPL 165
 INGPLEX SDUMP 166
 INGPLEX SVCDUMP 168
 INGPLEX SYStem 143
 INGPOST 220, 234
 INGRELS 76
 INGREQ 89
 precheck option for DB2 93
 INGREQ verification 92
 INGRPT 85
 INGRTCMD 229
 INGSCHED 97, 195
 INSEND command 185
 INGSET 32, 33
 INGTHRES 180, 183
 INGTHRES command 180
 INGTPO 225, 233
 INGTRIG 99
 INGVOTE 95, 192
 Initialize automation flag 31
 initializing SA z/OS 53
 integrated system channel 10
 interested operator list 118
 adding your ID 118
 displaying list 118
 join or leave 118
 ISPF dialog 204, 263
 ISQ900I message 118
 ISQ901I message 118
 ISQCMON 118
 ISQEHARP
 target hardware summary panel 208
 ISQEPNVC
 PATH detail panel 211
 ISQESUM
 status summary panel 208
 ISQETARG
 target system summary panel 210
 ISQETRGG
 guests summary panel 210
 ISQHELP command 69
 ISQSTART 58
 ISQSTOP 58
 ISQTRACE command 212
 ISQVARS command
 change status
 target hardware 129
 target system 127
 target system attention 130
 ISQXDST 127

ISQXDST command 118, 119
 ISQXMON 118
 issuing commands 67

L

licensed documents xvi
 logical partition 6
 LookAt message retrieval tool xvi
 lost&found anchors 222
 LPAR mode 7
 LU 6.2
 routing commands 184

M

Main Menu 68
 major resources on the NMC 220
 mapping
 NetView alerts to Tivoli events 249
 SA z/OS messages and Tivoli
 events 243
 mapping statuses
 automation agent to automation
 manager, 47
 marked and unmarked resources 236
 message classes 119
 message retrieval tool, LookAt xvi
 messages 243
 interested operator list 118
 monitoring 117
 notification operators 119
 waiting for IEA101A start
 message 129
 waiting for IEA347A start
 message 129
 waiting for VM start message 129
 minor resources on the NMC 220
 monitor resources
 using the NCCF panels 124
 using the NMC 125
 monitoring
 communications paths 131
 hardware 122
 health performance 124
 messages 117
 outstanding WTORs 117
 processors 126
 with SDF 133
 monitoring alerts 133
 MOVE 108
 move functions 22
 move group start type 52
 move group stop type 52
 MOVED 135
 MOVED automation status 42
 moving resources 109

N

names for processor operations
 objects 232
 names for system operations objects 231
 nature of application groups
 BASIC 108
 MOVE 108

nature of application groups (*continued*)
 SERVER 108
 NCCF panels
 using with monitor resources 124
 NetView
 automation procedure 11
 commands
 LIST ASSIGN=GROUP 119
 RMTCMD 184
 ROUTE 184
 logging on 53
 RMTCMD routing 184
 starting 57
 terminating after recovery
 actions 283
 NetView 3270 Management Console 254
 NetView alerts
 mapping to Tivoli events 249
 NetView automation table 11
 identifying 62
 NetView connection path 7, 131
 displaying status 131
 NVC 131
 SNMP 131
 NetView Management Console 4, 67,
 217
 creating views 230
 introduction 217
 issuing commands 226
 marked and unmarked resources 236
 minor resources 237
 monitoring 219
 names for processor operations
 objects 232
 names for system operations
 objects 231
 problem management 235
 processor operations 234
 properties of objects 223
 starting 218
 network views 230
 NMC 4, 217
 creating views 230
 introduction 217
 issuing commands 226
 marked and unmarked resources 236
 minor resources 237
 monitoring 219
 names for system operations
 objects 231, 232
 problem management 235
 processor operations 234
 properties of objects 223
 starting 218
 using with monitor resources 125
 NMC workstation 8
 non-disruptive restoration 27
 notification operators 119
 adding 121
 deleting 122
 settings 120
 notify messages 120
 NVC (NetView connection) 7

O

- objects 11
 - grouping 12
- observed status 33, 34
- observed statuses 38
- OCF (operations command facility) 6
- OCF-based processor 6
- online help 69
- operation mode 113
- operator console 204
- order 18
- OSA (open system adapter) 7
- outstanding WTORs
 - displaying 117
 - SDF colors 136
- override
 - resource 99
 - schedule 98
- override settings, displaying 105
- overrides 14, 15
- overriding schedules 196
- overview of SA z/OS 3

P

- PAM 18
- panels
 - DISPMTR 125
 - INGHC 176
 - INGSEND 185
 - INGTHRES 180
 - INGTHRES (Set Thresholds) 180
 - Path Detail 131
 - SETSTATE 117
 - Target Hardware Summary 123
 - TIMER Display 113
- parallel sysplex 7
- partitioning
 - logical 7
 - physical 7
- PATH detail panel ISQEPNVC 211
- pending policy changes 157
- physically partitioned configuration 7
- policy 8
- policy database 11
 - sharing 273
- port
 - ESCON port 8
- precheck option for DB2 93
- preference
 - base preference 24
 - calculated preference 24
 - effective preference 24
 - preference adjustment 24
- preference bonuses 26
- preference scenarios 26
- preference values 23, 25
- primary automation manager 18, 54
- priorities of subsystems 135
- priority of requests
 - FORCE 15
 - HIGH 15
 - LOW 15
- problem determination 273
- problem management on the NMC 235
- processor 6

- processor (*continued*)
 - monitoring status 126
 - OCF-based 6
 - target hardware summary 123
- processor control file 11
- processor operations 4
 - guest machines support 207
 - starting 58
 - stopping 58
- processor operations commands
 - ISQTRACE 212
- processor operations on the NMC 234
- processor operations status
 - information 122
- processors 10
- ProcOps
 - First Failure Data Capture support 279
 - tracing and debugging 278
 - tracing specific connections 279
 - using AOCTRACE 278
- ProcOps dialog 207
- propagation of conflicting requests 21
- propagation of requests 19
- properties of objects on NMC 223
- PSM console mode 211
- PSM control program
 - restarting 214
 - starting 212
 - stopping 213

R

- rebuild 157
 - system-managed 147, 157
 - user-managed 147, 157
- recovery actions
 - BCP Internal Interface 281
 - terminating NetView 283
 - terminating the BCP Internal Interface 283
- Recovery automation flag 31
- recovery mode 25
- refreshing automation policy 61
- refreshing the automation manager
 - configuration file 62
- regular monitoring 49
- regular resources 219
- relationship
 - start 19
 - stop 19
- relationship information 76
- relationships concept 20
- reloading automation control file 61
- reloading the NetView automation table 61
- repair, takeover file 55
- reports
 - HealthChecker 175
 - HealthChecker, about 175
 - HealthChecker, recommended procedure 176
 - HealthChecker, types of actions 175
- request 18, 95
- request propagation 19
- requests
 - creation of 14

- requests (*continued*)
 - hierarchy 15
- requests and goals 19
- resource
 - depending 19
 - supporting 19
- resource hierarchy with SDF 140
- resource information
 - sysplex-wide 71
- resource override 97, 99
- resource overrides 14
- resource subtype 74
- resources 11
 - start and stop 89
- Restart automation flag 32
- RESTART automation status 43
- restarting the PSM control program 214
- restoration
 - disruptive 27
 - non-disruptive 27
- Restructured Extended Executor (REXX) 11
- resynchronization 50
- REXX procedure 11
- RMTCMD command 184
- RODM (resource object data manager) 8
- ROUTE command 184
- routing
 - commands 184
- RUNNING automation status 43

S

- SA z/OS
 - overview 3
 - starting 53
- SA z/OS components
 - I/O operations 5
 - processor operations 4
 - system operations 4
- SA z/OS event 242, 243
- SA z/OS main menu 68
- SA z/OS messages 243
 - mapping to Tivoli events 243
- SA z/OS topology manager
 - behavior 225
 - initialization 225
- SA z/OS topology manager commands
 - INGPOST 234
 - INGTOPO 233
- SA z/OS topology manager
 - overview 217
- SAM 18
- scenarios 191
- schedule 15, 97
- schedule override 97, 98, 196
- schedule overrides 15
- SDF
 - monitoring application groups 140
 - monitoring with 133
 - monitoring WTORs 140
 - resource hierarchy 140
- SDF (status display facility) 12
 - customizing 133
 - how it works 137
- SDF command 133, 140, 141
- SDF hierarchy 141

- SDF, using 136
- second level systems
 - VM support 207
- secondary automation manager 18
- SERVER 108
- server group start type 52
- server group stop type 52
- server management 22
- service period 97
- service period overrides 15
- SETSTATE 114
- SETSTATE command 115
- SETTIMER 112
- setting automation on or off 106, 107
- Shutdown automation flag 32
- single point of control 184, 186
- single system image 184, 186
- SLIP TRAP 171
- specific connections, tracing 279
- SPFEDIT 273
- start resources 89
- start type
 - basic group 51
 - move group 52
 - server group 52
 - using group 51
- startability status 33, 36
- STARTED automation status 43
- STARTED2 automation status 44
- starting
 - applications 58
 - automation manager 53
 - I/O operations 57
 - NetView 57
 - processor operations 58
 - PSM control program 212
 - SA z/OS 53
 - subsystems 93
 - system operations 57
- status
 - ABENDING 38
 - ACTIVE 39
 - application monitor status 45
 - ASSIST 39
 - AUTODOWN 39
 - Automatic Restart Manager 46
 - automation agent status 38
 - AUTOTERM 40
 - BREAKING 40
 - BROKEN 40
 - clear to send 132
 - closed
 - target system 128
 - CTLDOWN 40
 - DCCF 130
 - determination 48
 - disabled wait 130
 - displaying with DISPSTAT 77
 - displaying with SDF 133
 - DOWN 41
 - ENDED 41
 - ENDING 41
 - EXTSTART 41
 - FALLBACK 42
 - HALFDOWN 42
 - HALTED 42
 - INACTIVE 42
 - status (*continued*)
 - initialized 128
 - IPL complete 128
 - IPL failed 128
 - last significant message 133
 - load failed 128
 - MOVED 42
 - NetView connection path 131
 - observed status 38
 - path problem 132
 - powered off 129
 - processor operations
 - attention status 130
 - path detail 131
 - target system attention status 130
 - target system status 128
 - processor operations path status 132
 - processors 126
 - RESTART 43
 - RUNNING 43
 - stage-1 activation complete 129
 - STARTED 43
 - STARTED2 44
 - STOPPED 44
 - STOPPING 44
 - STUCK 44
 - target hardware problem 129
 - target system status 128
 - UP 45
 - waiting for IEA101A start message 129
 - waiting for IEA347A start message 129
 - waiting for VM start message 129
 - ZOMBIE 45
 - Status Display Facility 133
 - status forwarding path
 - failure 273
 - status of a resource
 - automation status 33
 - compound status 33
 - desired status 33
 - observed status 33
 - startability status 33
 - status of resources 33
 - status panels 77
 - status summary 126
 - status summary panel ISQESUM 208
 - statuses
 - mapping automation agent to observed 47
 - statuses supplied by the automation agent
 - application monitor status 38
 - Automatic Restart Manager status 38
 - automation agent status 38
 - stop resources 89
 - stop type
 - basic group 51
 - move group 52
 - server group 52
 - using group 51
 - STOPPED automation status 44
 - stopping
 - applications 58
 - automation manager 54
 - I/O operations 57
 - stopping (*continued*)
 - processor operations 58
 - PSM control program 213
 - subsystems 94
 - STOPPING automation status 44
 - structure
 - duplexing
 - system-managed 158
 - user-managed 158
 - pending policy changes 157
 - rebuild 157
 - system-managed 147, 157
 - user-managed 147, 157
 - STUCK automation status 44
 - subsystem colors and priorities 135
 - subsystem priorities 135
 - subsystems, stopping 94
 - subtype 74
 - switch 202
 - switch matrix 266
 - switches 8
 - switching
 - alternate CDS to primary CDS 160
 - CFRM policy 162
 - SYSDSN 273
 - sysplex 7
 - monitor and control 141
 - sysplex automation 141
 - sysplex timer 8
 - sysplex-wide resource information 71
 - system operations 4
 - starting 57
 - system operations control files
 - automation control file 8
 - automation manager configuration file 8
 - System Operations main menu 68
 - system-managed duplexing 158
 - system-managed rebuild 147, 157
 - systems
 - grouping 12
 - monitoring status 127

T

- TAF (Terminal Access Facility)
 - fullscreen sessions 188
 - panels 189
 - TAF fullscreen display 189
- takeover file repair 55
- takeover file, dynamic 56
- target hardware
 - status 129
- target hardware summary 123
- target hardware summary panel ISQEHARP 208
- target processors 6
 - monitoring status 126
- target system
 - status 127
 - status panels
 - interested operator list 119
 - overall status 126
 - status summary 126
 - summary status 127
- target system summary 127

- target system summary panel
 - ISQETARG 210
- target systems
 - monitoring messages 117
 - monitoring status 127
- TEC 242
- TEC event server 242
- TEC Notification by SA z/OS 241
 - operation 251
 - starting 251
 - termination 256
- TEC rules 249
- terminal access facility fullscreen
 - sessions 188
- threshold
 - for z/OS components 179
- threshold settings 180
- thresholds 179
 - set or change 183
- timer
 - create 112
 - delete 113
 - modify 113
 - NetView 112
- TIMERS command 112
- Tivoli administrator 241
- Tivoli desktop 241
- Tivoli Enterprise Console 242
- Tivoli events 242
- tokens 220, 223
- trace records, printing 276
- trace services for the automation manager 274
- tracing 274
 - BCP Internal Interface 278
 - ProcOps 278
- tracing specific connections 279
- trigger condition list 199
- trigger information 99
- tutorial
 - I/O operations ISPF dialog 271

U

- UP automation status 45
- user IDs
 - for starting SA z/OS 53
- user-managed duplexing 158
- user-managed rebuild 147, 157
- using
 - NCCF panels with monitor resources 124
 - NMC with monitor resources 125
- using group start and stop types 51

V

- verify INGREQ 92
- views
 - creating 230
 - exception views 230
 - network views 230
- VM
 - second level systems support 207
- vote 18, 95

W

- WTORs
 - displaying in z/OS system log 117
 - displaying outstanding WTORs 117
 - hardware 117
 - monitoring with SDF 133, 140
 - SDF colors 136

Z

- z/OS operator console 204
- ZOMBIE automation status 45

Readers' Comments — We'd Like to Hear from You

System Automation for z/OS
User's Guide
Version 2 Release 3

Publication No. SC33-7040-08

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? Yes No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.



Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Deutschland Entwicklung GmbH
Department 3248
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany



Fold and Tape

Please do not staple

Fold and Tape



Program Number: 5645-006

Printed in USA

SC33-7040-08



Spine information:



System Automation for z/OS

User's Guide

Version 2 Release 3

SC33-7040-08