Unicenter

Automation Services Common User Guide

12th edition

P01-187

Booklist

Computer Associates
The Software That Manages eBusiness

Edition	Publication Number	Product Version	Min. MS Level	Publish Date
9th Edition	P01-187	5.0	5.0	September 2001

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. ("CA") at any time.

This documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of CA. This documentation is proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this documentation for their own internal use, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to return to CA the reproduced copies or to certify to CA that same have been destroyed.

To the extent permitted by applicable law, CA provides this documentation "as is" without warranty of any kind, including without limitation, any implied warranties of merchantability, fitness for a particular purpose or noninfringement. In no event will CA be liable to the end user or any third party for any loss or damage, direct or indirect, from the use of this documentation, including without limitation, lost profits, business interruption, goodwill, or lost data, even if CA is expressly advised of such loss or damage.

The use of any product referenced in this documentation and this documentation is governed by the end user's applicable license agreement.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227-7013(c)(1)(ii) or applicable successor provisions.

© 2001 Computer Associates International, Inc.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Table of Contents

Part I Overview

Chapter 1 Introduction 1-1

What Automation Services Offers 1-2
Multisystem Support 1-2
Basic Features 1-3

How the Products Help You Manage Your Network and System Resources 1-3

Benefits 1-4

About this Guide 1-4

What You Should Know Already 1-5

Product Specific Information 1-5

Chapter 2 Concepts 2-1

Top-Down Approach to Operations 2-2
Automation Services Approach 2-4

Single System or Multisystem 2-5

Services 2-5

System Images and Versions 2-6
Version Control 2-6

Knowledge Base Components 2-7

Part II Service Management

Chapter 3 About Service Management 3-1

Managing Services 3-2

Features 3-2
AutoAssist 3-3
ResourceView 3-3
Automation Engine 3-3
Desired State Management 3-4
Startup, Shutdown, and Recovery 3-4
ServiceView 3-5
Operation States 3-6
Desired States 3-6
Actual States 3-6
Logical States 3-7
Operation Modes 3-8
Global Operation Mode 3-8
Monitor Features 3-9
Generic Commands 3-9
Availability Maps 3-9
Processes 3-9
Reporting 3-10
Knowledge Base 3-11
Multisystem Support 3-11
Peer-based Automation 3-12
Multisystem Knowledge Base 3-12
Management of Services 3-12
Management Services 3-13

Implementing the Operations Environment 3-13
Managing the Desired State of Resources 3-13
Providing a Service-driven Operations Perspective of Your Networks and Systems 3-14
Multisystem Consideration 3-14
Typical Implementation Tasks 3-14
Task 1—Implement Automation Services to Monitor Resources 3-15
Task 2—Implement Automation Services to Automate Operations 3-16
Task 3—Implement Automation Services to Manage Business Functions 3-17

Developing System Images That Contain the Resources You Want to Manage 3-18
Implementing a System Image 3-18
System Image Components 3-20

Defining Services 3-21

Specifying the Availability of Services and Their Members 3-21
Activation of a Service 3-22
Inactivation of a Service 3-22
Status of a Service 3-22

Monitoring Service-driven Operations 3-23
Status Monitor Filter 3-23
Graphical Monitor Icon Panel 3-23

Chapter 4 Working With System Image Definitions 4-1

When Do You Need to Define System Images? 4-2

Implementing System Images 4-2
Example 1 4-3
Example 2 4-3

Accessing System Image Definitions 4-3

Actions That You Can Perform on a System Image 4-4

Defining a System Image 4-4
Example 4-5

Maintaining System Images 4-6
Merging System Images 4-6

Chapter 5 Adding Resource Definitions to a System Image 5-1

Why Do You Need to Define Resources? 5-2

Implementing Resource Definitions 5-2
Using Templates 5-2
Using the USRCLS Class 5-2

Accessing Resource Definitions 5-3
Accessing Resource Definitions Through the Resource Definition Panel 5-3
Accessing Resource Definitions Through the System Image List Panel 5-3

Actions That You Can Perform on a Resource Definition 5-4

Defining a Resource to a System Image 5-5
Describing the Resource 5-7
Using Resource Templates 5-9
Controlling the VTAM ACB of a Resource 5-9
Specifying the Operation Mode 5-9
Defining the Availability of the Resource 5-10
Defining the Activation Details 5-11
Defining the Restart Control 5-14
Defining the Inactivation Details 5-17
Defining the Forced Inactivation Details 5-19
Defining the Display and Heartbeat Details 5-22
Defining the Status Monitor Message Details 5-24
Extended Message Definition Considerations 5-25
Using the RECOVERED Actual State 5-26
Desired State Management Considerations 5-27
Implementing the State Change Exits 5-28
Defining the Logging Details 5-29
Specifying the Owner Details 5-31
Implementing the Extended Function Exit 5-31

Handling Timeouts 5-32

Specifying Messages in a Resource Definition 5-33
Using Messages in the Transient Log 5-34
Using Special Message Prefixes 5-35

Extending the Definition of Resource Message Rules 5-36
Defining an Extended Message Filter..... 5-36
Defining Event-related Actions 5-38
Implementing Event Exits 5-38
Defining Extended Display Attributes..... 5-39
Overriding the Status Monitor Fields 5-40
Overriding the Display Attributes 5-40
Using the User-defined Keyword 5-41
Using User-defined Variables 5-41
Documenting the Message Rule 5-41

Executing an NCL Procedure..... 5-42

Creating User-defined Resource Subclasses 5-43

Using Logical Resources..... 5-44
Example 5-45

Defining Resource Relationships 5-46
Primary and Alternate Relationships 5-48
Effect of Resource Relationships on Operations 5-48
Starting a System That Contains Only Automated Resources . 5-48
Stopping a System That Contains Only Automated Resources..... 5-48
Starting a System That Contains Automated and Manual Resources..... 5-48
Stopping a System That Contains Automated and Manual Resources..... 5-49
Effect of a Resource Set to the OFF Operation Mode on a Relationship 5-49

Implementing Resource Definitions for System Load Balancing..... 5-50

Maintaining Resource Definitions..... 5-51

Chapter 6 Defining and Maintaining Services 6-1

Accessing Service Definitions 6-2

Actions That You Can Perform on a Service Definition 6-2

Defining a Service 6-3
- Describing the Service 6-4
  - Specifying the Operation Mode 6-4
- Defining the Availability of the Service 6-5
- Selecting Service Members 6-6
  - Defining the Weight of a Service Member 6-7
  - Viewing the Service as Defined by the Service Filters 6-7
- Defining the State Thresholds 6-8
  - Example 6-9
- Implementing the State Change Exits 6-10
- Defining the Logging Details 6-10
- Specifying the Owner Details 6-10
- Implementing the Extended Function Exit 6-10

Maintaining Service Definitions 6-11

Backing Up Service Definitions 6-11
- Updating Service Definitions in a Backup Service Image 6-13
- Restoring a Service Definition from a Backup Service Image 6-13
- Merging Two Service Images 6-15

Chapter 7 Defining and Maintaining Availability Maps 7-1

Implementing Availability Maps..... 7-2
Rules for Availability Map Definitions 7-3

Accessing Availability Map Definitions 7-3
Temporary Availability Maps..... 7-3

Actions That You Can Perform on an Availability Map Definition 7-4

Defining an Availability Map..... 7-4
Defining Timers 7-5
Scheduling Processes..... 7-5
Resetting Manual Overrides 7-5
Example 7-6

Viewing Timer Information 7-7
Viewing All Timer Information..... 7-7
Viewing the Timer Information in One Availability Map..... 7-7

Attaching a Service or Resource Definition to an Availability Map ... 7-8
Example 7-9

Detaching Service or Resource Definitions From an Availability Map..... 7-10
Detaching Services or Resources From a List 7-10
Detaching a Service or Resource While You Are Working on Its Definition 7-10

Maintaining Availability Map Definitions 7-10

Chapter 8 Defining Processes to Implement Complex Operations 8-1

Implementing Processes 8-2
Process Types 8-3

Accessing Process Definitions 8-5

Actions That You Can Perform on a Process Definition 8-5

Defining a Process 8-6
Setting Macro Parameters 8-8
Using a Variable as a Macro Parameter 8-9
Example—Defining a Display Process 8-9

Testing a Process 8-15
Testing a Process by Execution as a Single Task 8-15
Testing a Process Interactively 8-15

Logging Process Activities 8-16

Maintaining Process Definitions 8-16

Backing Up Global Processes 8-16
Updating Global Process Definitions in a Backup Global Process Image 8-18
Restoring a Global Process Definition From a Backup Global Process Image 8-19
Changing a Global Process to a Local Process 8-20
Merging Two Global Process Images 8-21

Chapter 9 Defining and Maintaining Resource Templates 9-1

Types of Resource Templates 9-2
Product Specific Resource Templates 9-2
USRCLS Class Template..... 9-2

Setting Up Your Template System..... 9-3
Merging $TEMPLAT System Images..... 9-3

Defining and Maintaining Resource Templates..... 9-4
Accessing the Resource Template Definitions 9-4
Defining a Resource Template 9-5
Using Variables 9-5
Defining Variables..... 9-5
Using Less Than Signs to Represent a Left-justified Fixed-length Variable Field Value..... 9-5
Using Greater Than Signs to Represent a Right-justified Fixed-length Variable Field Value 9-5
Maintaining Resource Template Definitions..... 9-6
Applying Updated Templates..... 9-6

Defining and Maintaining Availability Maps in a Template System Image 9-7
Accessing the Availability Map Definitions in a Template System Image..... 9-7
Defining an Availability Map in a Template System Image 9-7
Maintaining the Availability Map Definitions in a Template System Image..... 9-7

Defining and Maintaining Processes in a Template System Image 9-8
Accessing the Process Definitions in a Template System Image .. 9-8
Defining a Process in a Template System Image 9-8
Maintaining the Processes in a Template System Image..... 9-8

Converting a Resource Definition Into a Resource Template..... 9-9

Chapter 10 Testing Definitions by Using Simulated Events 10-1

What Is the Event Simulator?..... 10-2

Generating Simulated Events 10-2
Defining a Simulated Event..... 10-3

Interpreting the Results of Event Simulation 10-4
Summarizing the Results 10-5

Maintaining Simulated Event Definitions..... 10-5

Chapter 11 Defining and Maintaining Filters for the Status Monitor 11-1

Implementing the Status Monitor Filters 11-2

Accessing Status Monitor Filter Definitions 11-2

Defining a Status Monitor Filter 11-3
- Using the Status Monitor Filter Panel 11-4
  - Status Monitor Views 11-4
- Defining the Status Monitor Filter Expression 11-5
- Example 11-5

Maintaining Status Monitor Filter Definitions 11-6

Chapter 12 Defining and Maintaining Resource Groups, Icons, and Icon Panels for the Graphical Monitor 12-1

Customizing the Graphical Monitor 12-2

Defining and Maintaining Resource Groups for Icons 12-2
- Accessing Resource Group Definitions 12-3
- Defining a Resource Group 12-3
  - Using the Resource Group Filter Definition Panel 12-5
  - Defining the Resource Group Filter Expression 12-5
- Maintaining Resource Group Definitions 12-6

Defining and Maintaining Icons 12-6
- Accessing the Icon Definitions 12-6
- Defining an Icon 12-7
  - Using the Icon Editor Panel 12-8
  - Editing the Icon 12-9
  - Example 12-13
- Maintaining Icon Definitions 12-14

Defining and Maintaining Icon Panels 12-15
Accessing Icon Panel Definitions 12-15
Defining and Generating an Icon Panel 12-16
Using the Icon Panel Generator Panel 12-17
Adding an Icon to the Icon Panel 12-18
Attaching a Resource Group to an Icon on the Icon Panel 12-19
Duplicating an Icon on the Icon Panel 12-19
Moving an Icon on the Icon Panel 12-19
Editing an Icon on the Icon Panel 12-20
Getting Information About an Icon on the Icon Panel 12-20
Deleting an Icon From the Icon Panel 12-21
Maintaining Icon Panel Definitions 12-21

Editing a Generated Icon Panel 12-22

Chapter 13 Monitoring and Controlling Services and Resources 13-1

Monitoring Services and Resources 13-2

Controlling Services and Resources 13-2
Working With Resources Registered to the Automatic Restart Manager 13-3

Using the Status Monitor 13-3
Accessing the Status Monitor 13-4
Interpreting the Status Monitor Display 13-5
Operating in the Status Monitor 13-7
Organizing the Information on the Status Monitor 13-7
Changing to a Different Status Monitor Filter 13-10
Changing Your Default Status Monitor Profile 13-11

Using the Graphical Monitor 13-12
Accessing the Graphical Monitor 13-13
Interpreting the Graphical Monitor Display 13-14
Operating From the Graphical Monitor 13-14
Using the Z (Zoom) Command 13-14
Using Other Commands 13-15
Changing to a Different Default Icon Panel 13-16

Using Commands 13-16
Finding Out More About Monitored Services and Resources 13-16
Finding Out About Relationships Between Monitored Resources 13-17
Removing the User-defined Extended Display 13-17
Accessing Extended Functions 13-17
Replying to a WTOR Message 13-18
Overriding the Operation Mode 13-18
Overriding the Service and Resource States 13-19
Overriding the Actual State 13-19
Overriding the Desired State 13-19
Stopping a Parent and Its Children 13-20
Browsing or Updating the Definition of Monitored Services and Resources 13-20
Modifying the Scheduled Operations Changes for a Monitored Item 13-21
Overriding the Scheduled Operations Changes 13-21
Browsing or Updating Availability Maps 13-21
Moving the Control of a Resource From One Region to Another 13-22
Controlling Manual Services and Resources 13-23
Starting Manual Services and Resources 13-23
Stopping Manual Services and Resources Normally 13-23
Stopping Manual Resources by Force 13-23

Recovering a Failed Resource 13-24

Recovering From Failed Automation 13-25

Acknowledging a Link Failure 13-26

Responding to the Initialization Status Panel 13-27

Responding to the Database Synchronization Panel 13-27

Using the Logs 13-28
Accessing an Activity Log 13-28
Browsing the Activity Log Online 13-28
Accessing a Transient Log 13-29
Setting Criteria to Display Logged Messages Selectively 13-29
Displaying User-defined Log Messages 13-31
Obtaining Help on a Logged Message 13-31
Printing a Transient Log 13-31
Resetting a Transient Log 13-31
Changing Logging Options 13-31
Merging Transient Logs 13-31
Transient Logs From Different Time Zones 13-32
Size of a Merged Transient Log 13-32

Obtaining Operations Statistics 13-32
Obtaining the Operations Statistics of a Monitored Service or Resource 13-32
Interpreting the Statistics 13-33

Chapter 14 Controlling the System Image 14-1

Changing the Global Operation Mode 14-2
Example 14-3
Loading a System Image 14-3
Swapping the Active System Image 14-6
Cold and Warm Load Features 14-7
Tuning the CPU Time Consumed by the Loading of a System Image 14-7
Shutting Down Resources in an Active System Image 14-8
Shutting Down Automated Resources Only 14-9
Shutting Down All Resources 14-10
Restarting Resources in an Active System Image 14-11
Effect of Checkpoint Restart on the STARTSYS Command 14-12
Behavior of the STARTSYS Command After a SHUTSYS Command 14-12
Behavior of the STARTSYS Command After a SHUTFORCE Command 14-12

Part III Event Management

Chapter 15 About Event Management 15-1

What is EventView? 15-2
Event-based Automation 15-3
Sample Message Suppression Rule Sets 15-3
Console Message Consolidation 15-3
Alert Monitoring 15-4
Benefits of Using EventView 15-4
Developing Event Management Rules 15-5
Selecting the Messages to Be Monitored 15-7
Message Consolidation Enabled 15-8
Message Consolidation Disabled 15-10
Generating Alerts 15-10

Chapter 16 Defining and Maintaining Rulesets..... 16-1

Working With Rulesets..... 16-2
Adding a Rule Set 16-3
Specifying Control Options 16-3
Monitoring Rule Set Status 16-4
Gathering Statistics 16-4
Changing the Rule Set Associated With a System Image..... 16-5

Adding Associated Rules..... 16-5
Adding Message Rules..... 16-6
Message Execution Conditions..... 16-6
Adding Timers 16-6
Grouping Messages..... 16-7
Adding Message Group Rules 16-7
Associating Message Rules With Message Group Rules 16-8

Adding Initial Actions 16-9
Execution of Initial Actions 16-10

Including Rule Sets in Other Rule Sets 16-11

Maintaining Rule Sets..... 16-11

Using EventView Variables 16-12
Viewing EventView Variables..... 16-12

Chapter 17 Processing Messages 17-1

Specifying Message Filtering Criteria 17-2
Using Wildcards in Message Text 17-3
Specifying Extended Filtering Criteria 17-4
Specifying Message Text Analysis Criteria 17-5
Linking Tests in an Expression 17-5
Testing EventView Variables 17-5
Specifying Execution Conditions 17-6
Dealing With Overlapping Rules 17-6

Specifying Message Delivery 17-7
Setting the Deliver Flag 17-7
Setting Delivery Thresholds 17-8
Specifying a Correlation Key 17-9

Modifying Messages 17-9
Specifying Replacement Text 17-10
Specifying System Message Presentation Parameters 17-10
Specifying SOLVE Message Presentation Parameters 17-10

Specifying Actions to be Taken in Response to Messages 17-11

Suppressing Messages 17-12

Chapter 18 Defining Timers 18-1

What Is a Timer Rule? 18-2

Adding and Maintaining Timers 18-2
Performing Catchup 18-3
Specifying a Catchup Window 18-3
Specifying Timer Schedule Items 18-4
Adding Further Schedule Items 18-5
Viewing the Next Execution Display for Timer Schedule Items 18-6
Updating Timer Schedule Items 18-6
Deleting Timer Schedule Items 18-6
Specifying Actions to be Taken When a Timer Item is Triggered 18-7
Example 18-7

Displaying Active Timer Rules 18-8

Chapter 19 Message Learning 19-1

What is Message Learning? 19-2

Controlling Message Learning 19-2

Browsing and Updating Learnt Messages 19-3

Generating a Rule for a Learnt Message 19-5

Resetting New Message Indicators 19-7

Deleting All Learnt Messages 19-7

Chapter 20 Defining and Maintaining Message Profiles for the Consolidated Console 20-1

How Console Consolidation Works in a Multisystem Environment.... 20-2

Implementing Message Profiles 20-3
- Rules for Defining and Using Message Profiles 20-4
  - Create New Message Profiles in a Single System First 20-4
  - Use Unique Message Profile Names and IDs 20-4
  - Using Wildcards 20-5
  - Using Ranges 20-6
  - Including and Excluding Information 20-7

Accessing the Message Profile Definitions 20-8

Defining a Message Profile 20-8
- Defining the Profile Details 20-9
- Specifying the System Criteria 20-9
- Specifying the Message ID Criteria 20-10
- Specifying the Job Criteria 20-10
- Specifying the System Codes Criteria 20-10
- Specifying the Message Type, Level, and Job Criteria 20-11
  - Message Types 20-11
  - Message Levels 20-11
  - Job Types 20-11
  - Job Classes 20-11
- Example 1—Profiling Specific Messages 20-12
- Example 2—Profiling Messages for Specific Jobs 20-15
- Example 3—Profiling All Messages 20-17
- Example 4—Profiling Messages for a Particular System 20-18

Changing the Activation Status of a Message Profile 20-18

Activating Message Profiles 20-19
Message Profile Size Considerations 20-19
Maintaining Message Profile Definitions 20-19

Chapter 21 Monitoring Messages 21-1

Message Monitor 21-2
Operator Profile 21-2
Window 21-3
Command Line 21-3
Roll Delete Area 21-4
Non-roll Delete Area 21-4
Title Line 21-4
Controlling the Roll Delete Message Display Flow 21-5
Handling Non-Roll Delete (NRD) Messages 21-5
Deleting NRD Messages 21-5
Removing NRD Messages (NRD Cursor Deletion) 21-6
Restoring NRD Messages 21-6
Prefixing Messages With the System Name 21-6
Consolidated Console Setup Requirements 21-7
Authorization Requirements 21-7
Monitor Profile Requirements 21-7
Accessing the Consolidated Console 21-8
If the Console Is Not Displaying System Messages... 21-8
Using Message Profiles to Select the Messages to Be Monitored 21-10
Replying to a WTOR Message From the Consolidated Console 21-11
Leaving the Consolidated Console 21-12

Chapter 22 Managing Alerts 22-1

Accessing the Alert Monitor 22-2
Sorting Alerts 22-3
Filtering Alerts 22-3
Change the Display Format 22-3

Working With Alerts 22-4
Viewing the Details of an Alert 22-4
Accessing the Status Monitor From the Alert Monitor 22-5
Accessing the Transient Log From the Alert Monitor 22-5
Raising a Problem Ticket for an Alert 22-5
Adding Operator Notes to an Alert 22-5
Closing an Alert 22-6

Displaying Alert History 22-6

Part IV Reference Material

Appendix A Commands A-1

Line Commands A-2
Line Command Parameters A-2

Primary Commands A-3
Primary Command Parameters A-3
GLOBAL Command Parameters A-3
LOAD Command Parameters A-3

Appendix B System Console Interface B-1

$RMCONS NCL Procedure B-2
Line Commands B-3
Primary Commands B-3

$RMCLOG NCL Procedure B-5

Appendix C Exit NCL Procedures C-1

Exit to the SOLVE:Problem Application C-2

Exit to the OIM Problem Management Application C-4

Appendix D Variables D-1

Knowledge Base Variables D-2

Status Variables D-6

Active System Image Variables D-8

Message Variables D-9

Appendix E Macros E-1

Supplied Macros E-2

AOMALERT E-5

CHAIN E-6

COMMAND E-8

COMPARE E-11

CONCAT E-14

DELALERT E-16

EVENT E-17

EVVARGET E-19

EVVARSET E-20

EXECNCL E-21

EXTRACT E-22

GENALERT E-27

GETSTAT E-30

GOTO E-32

PARSE E-33

REPLY E-36

RUNPRCSS E-40

SETRC E-41

SETSTATE E-42

SETVARS E-45

SMFWRITE E-46

SNMPTRAP E-47

STARTNCL E-49

STOP E-50

SUBJOB E-51
SUBPRCSS..... E-54
SVAPI..... E-56
SVCMD E-59
SYSCMD..... E-60
TRANS E-63
WAIT..... E-65
WAITEVNT E-66
WAITSTAT..... E-69
WTO E-72
WTOR..... E-73

Appendix F Message Text Analysis F-1

Using Message Text Analysis F-2
Examples F-4
Example of the Use of the CT Operator F-4
Example of the Use of the EQ Operator F-5
Example of the Use of the GE Operator F-5
Example of the Use of the GT Operator F-6
Example of the Use of the LE Operator..... F-6
Example of the Use of the LT Operator..... F-7
Example of the Use of the NE Operator F-7
Linking Tests in an Expression..... F-8

Appendix G Boolean Operators G-1

Relational Operators G-2
Logical Operators G-3

Appendix H Knowledge Base Definition Classes H-1

Definition Classes..... H-2

Appendix I Reporting I-1

About Reporting I-2

Generating a Report..... I-3
Searching for Macros..... I-4
Searching for Message Rules..... I-6
Searching for Processes I-9
Using Your Search Output..... I-11
Sorting Your Output I-11
Locating Output..... I-11

Appendix J Message Routing Codes J-1

Message Routing Codes J-2

Appendix K Message Descriptor Codes K-1

Message Descriptor Codes K-2

Appendix L Message Levels..... L-1

Message Levels L-2

Glossary

Index

Figures and Tables

Figure 2-1. Operations Environment Model 2-3
Figure 2-2. Automation Services Knowledge Base Components 2-8
Figure 3-1. Implementing a System Image 3-19
Figure 4-1. Example—System Image Definition Panel 4-5
Figure 4-2. Merging System Images 4-7
Figure 5-1. Resource Definition—Main Features 5-6
Figure 5-2. Example—General Description Panel 5-7
Figure 5-3. Example—Availability Map Panel 5-11
Figure 5-4. Resource Starting Operation 5-12
Figure 5-5. Example—Activation Details Panel 5-14
Figure 5-6. Effect of Activation Retry Limits on Resource Restarts 5-15
Figure 5-7. Example—Restart Control Parameters Panel 5-16
Figure 5-8. Example—Inactivation Details Panel 5-19
Figure 5-9. Example—Force Inactivation Details Panel 5-21
Figure 5-10. Example—Display and Heartbeat Details Panel 5-23
Figure 5-11. Example—Status Monitor Message Details Panel 5-25
Figure 5-12. Using the RECOVERED Actual State 5-26
Figure 5-13. Example—State Change Exits Panel 5-28
Figure 5-14. Example—Automation Log Details Panel 5-29
Figure 5-15. Example—Defining Message Filters: 1 5-37
Figure 5-16. Example—Defining Message Filters: 2 5-37
Figure 5-17. Example—Define Extended Display Attribute Panel 5-39
Figure 5-18. DB2 Resource Relationships 5-45
Figure 5-19. Example—Relating Resources 5-47
Figure 5-20. Effect of a Resource Set to the OFF Operation Mode on a Relationship 5-49

Figure 6-1. Copying a Service Image 6-12
Figure 6-2. Creating a Backup Service Image 6-12
Figure 6-3. Restoring a Service Definition 6-14
Figure 6-4. Confirm Copy Replace Panel 6-14
Figure 6-5. Confirm System Image Merge Panel 6-15
Figure 7-1. Example—Availability Map 7-6
Figure 7-2. Example—Service Definition Availability Map Panel 7-9
Figure 8-1. Defining a Process 8-3
Figure 8-2. Process Search Path 8-4
Figure 8-3. Example—Process 8-7
Figure 8-4. The Parameters for STEP1 8-8
Figure 8-5. Example—Printer Display Process Steps 8-9
Figure 8-6. Process Step—Checking Printer Status 8-10
Figure 8-7. Process Step—Setting Printer Actual State 8-11
Figure 8-8. Process Step—Checking Whether Printing Is in Progress or Completed 8-12
Figure 8-9. Process Step—Extracting the Name of the Job Being Printed . 8-13
Figure 8-10. Process Step—Parsing the Extracted Name 8-13
Figure 8-11. Process Step—Displaying the Status of a Printing Job 8-14
Figure 8-12. Copying a Global Process Image 8-17
Figure 8-13. Creating a Backup Global Process Image 8-17
Figure 8-14. Restoring a Global Process Definition 8-19
Figure 8-15. Confirm Copy Replace Panel 8-20
Figure 8-16. Confirm System Image Merge Panel for Global Process Images 8-21
Figure 9-1. Applying a Template to Resources in a System Image 9-6
Figure 10-1. Defining a Simulated Event 10-3
Figure 10-2. Event Simulation Results 10-4
Figure 10-3. Event Simulation Results Summary 10-5
Figure 11-1. Status Monitor Filter List Panel 11-2
Figure 11-2. Status Monitor Filter Panel 11-4
Figure 11-3. Example—Service Filter Definition for the Status Monitor 11-5
Figure 12-1. Resource Groups List Panel 12-3
Figure 12-2. Resource Group Filter Definition Panel 12-5
Figure 12-3. Icon List Panel 12-6
Figure 12-4. Icon Editor Panel 12-8
Figure 12-5. Displayed Icon 12-13
Figure 12-6. Example—Icon Definition 12-14
Figure 12-7. Icon Panel Definition List Panel 12-15
Figure 12-8. Icon Panel Generator Panel 12-17
Figure 12-9. Panel List Panel 12-22
Figure 13-1. A Status Monitor Screen 13-4
Figure 13-2. The Graphical Monitor Screen 13-13

Figure 13-3. An Icon 13-14
Figure 13-4. Execute ACKLNKFAIL Command Panel 13-26
Figure 13-5. Statistics Panel 13-33
Figure 14-1. LOAD Command Parameter Specification Panel 14-4
Figure 14-2. Confirmation Panel 14-5
Figure 14-3. Initialization Status Panel 14-6
Figure 15-1. EventView Processing 15-2
Figure 15-2. Implementing a Rule Set 15-6
Figure 15-3. Console Message Consolidation 15-8
Figure 15-4. Implementing Message Profiles 15-9
Figure 15-5. Console Message Consolidation Disabled 15-10
Figure 16-1. Adding a Rule Set 16-3
Figure 16-2. Monitoring Rule Set Status 16-4
Figure 16-3. Associating Messages and Message Groups 16-8
Figure 16-4. Specifying Initial Actions 16-9
Figure 16-5. Setting EventView Variables 16-10
Figure 17-1. Specifying Extended Filtering Criteria 17-2
Figure 17-2. Specifying Execution Conditions 17-6
Figure 17-3. Specifying Message Thresholds 17-9
Figure 17-4. Specifying How to Modify a Message 17-10
Figure 17-5. Specifying How to Respond to a Message 17-11
Figure 18-1. Adding Timer Schedule Items 18-3
Figure 18-2. Full-screen Schedule Display 18-5
Figure 18-3. Displaying the Next Execution Times and Dates for a Timer Rule 18-6
Figure 18-4. Specifying Timer Actions 18-7
Figure 18-5. Displaying Active Timers 18-8
Figure 19-1. Browsing a Learnt Message 19-4
Figure 19-2. Generating a Message Rule 19-6
Figure 20-1. Multisystem Links for Consolidated Consoles 20-2
Figure 20-2. Example—Profile Details Panel 20-12
Figure 20-3. Example—System Specification Panel 20-13
Figure 20-4. Example—Message Specification Panel 20-14
Figure 20-5. Example—Job Name Specification Panel 20-14
Figure 20-6. Example—Message Job Specification Panel 20-15
Figure 20-7. Example—Profiling for Messages of Specific Jobs 20-16
Figure 20-8. Example—Profiling for All Messages 20-17
Figure 20-9. Example—Profiling for Messages on a Particular System 20-18
Figure 21-1. Message Monitor Window Layout 21-3
Figure 21-2. Private Message Profile Control Panel 21-10
Figure 22-1. Alert Monitor Panels 22-2
Figure D-1. Example—Using Knowledge Base Variables D-2
Figure D-2. Example—Using Message Variables D-9

Figure E-1. CHAIN Macro Parameter Definition (Example A) E-6
Figure E-2. CHAIN Macro Parameter Definition (Example B) E-7
Figure E-3. COMMAND Macro Parameter Definition E-10
Figure E-4. Extended Message Filtering Panel E-10
Figure E-5. COMPARE Macro Parameter Definition E-13
Figure E-6. CONCAT Macro Parameter Definition E-15
Figure E-7. DELALERT Macro Parameter Definition E-16
Figure E-8. EVENT Macro Parameter Definition E-18
Figure E-9. EVVARGET Macro Parameter Definition E-19
Figure E-10. EVVARSET Macro Parameter Definition E-20
Figure E-11. EXECNCL Macro Parameter Definition E-21
Figure E-12. EXTRACT Macro Parameter Definition (Example A) E-24
Figure E-13. EXTRACT Macro Parameter Definition (Example B) E-25
Figure E-14. EXTRACT Macro Parameter Definition (Example C) E-26
Figure E-15. GENALERT Macro Parameter Definition E-29
Figure E-16. GETSTAT Macro Parameter Definition E-31
Figure E-17. GOTO Macro Parameter Definition E-32
Figure E-18. PARSE Macro Parameter Definition E-35
Figure E-19. REPLY Macro Parameter Definition E-39
Figure E-20. RUNPRCSS Macro Parameter Definition E-40
Figure E-21. SETRC Macro Parameter Definition E-41
Figure E-22. SETSTATE Macro Parameter Definition E-44
Figure E-23. SETVARS Macro Parameter Definition E-45
Figure E-24. SMFWRITE Macro Parameter Definition E-46
Figure E-25. SNMPTRAP Macro Parameter Definition E-48
Figure E-26. STARTNCL Macro Parameter Definition E-49
Figure E-27. STOP Macro Parameter Definition E-50
Figure E-28. SUBJOB Macro Parameter Definition E-53
Figure E-29. SUBPRCSS Macro Parameter Definition E-55
Figure E-30. SVAPI Macro Parameter Definition E-58
Figure E-31. SVCMD Macro Parameter Definition E-59
Figure E-32. SYSCMD Macro Parameter Definition E-62
Figure E-33. TRANS Macro Parameter Definition E-64
Figure E-34. WAIT Macro Parameter Definition E-65
Figure E-35. WAITEVNT Macro Parameter Definition E-68
Figure E-36. WAITSTAT Macro Parameter Definition E-71
Figure E-37. WTO Macro Parameter Definition E-72
Figure E-38. WTOR Issue Macro Parameter Definition E-74
Figure I-1. Reporting Menu I-3
Figure I-2. Criteria List for Report Searches I-3
Figure I-3. Search for Macro Panel I-4
Figure I-4. Macro Search Result List I-5
Figure I-5. Search for Message Rule Panel I-6

Figure I-6. Message Rule Search Result List I-8
Figure I-7. Search for Process Panel..... I-9
Figure I-8. Process Search Result List..... I-10
Figure I-9. Sort Criteria Panel I-11

Table 3-1. Operation Modes 3-8
Table 5-1. Log Options 5-29
Table 6-1. Effect of Operation Mode on a Service 6-4
Table 13-1. Override Flags 13-6
Table 13-2. Severity Levels of the Messages in the Transient Log..... 13-30
Table 14-1. Global Operation Modes 14-2
Table D-1. Knowledge Base Variables..... D-2
Table D-2. Status Variables D-6
Table D-3. Message Variables..... D-9
Table E-1. Macros Used to Define Processes E-2
Table H-1. Knowledge Base Definition Classes..... H-2

Part I

Overview

1

Introduction

The key issue in operations is to maximize service availability for end users. As networks and systems become increasingly diverse and complex, the challenge for an enterprise is to ensure the availability of critical services and resources when required. Automation Services helps you operate, and maintain the availability of, large, centralized systems, or systems distributed over networks.

Note

Some products (for example, NetMaster for SNA and NetSpy) uses only a subset of these services. See the product guides.

This chapter contains the following topics:

- What Automation Services Offers
- How the Products Help You Manage Your Network and System Resources
- Benefits
- About this Guide
- What You Should Know Already
- Product Specific Information

What Automation Services Offers

Automation Services supports a suite of operations automation and management application products. These products support various network and system resources. These products can be implemented on one or more computers in an organization, depending on the operations requirements of the organization.

Automation Services enables you to automate the operation of the information system (IS) resources in your organization. It establishes a *region* that uses the information in a knowledge base to automate the operations processes in your organization.

The knowledge base contains the operations policies and methods for the delivery of services to your organization. By using the knowledge base, the region maintains the availability of the resources that deliver services to your organization. The operators are relieved of routine operations tasks, freeing them to concentrate on abnormal operations conditions.

For example, your organization might need to make a production system available to users at 0800 hours each weekday, bring the system down at 2300 hours to start backups, then make the production system available again at 0800 hours the next business day. The region can automate this and many other operations activities.

The region also minimizes the impact of a failure to the users of a service. If a failure occurs while the service is idle, the service is restarted automatically before a user requires that service. That is, the users are unaware of the failure and see the service as continuously available.

Note

Some products do not use the automation feature. These products use Automation Services to provide only knowledge base management and presentation services.

Multisystem Support

By installing the products on each system you want to manage, you can obtain a multisystem view of your resources from a single terminal.

Basic Features

Automation Services uses a range of mechanisms to automate your operations environment, including:

- Peer-to-peer architecture
- A powerful automation engine
- Time windows that indicate the availability of services and resources
- A multisystem knowledge base
- Comprehensive monitoring by using:
  - Line-by-line display of status information
  - Graphical display of status information
  - Messages display
  - Alerts display

How the Products Help You Manage Your Network and System Resources

The products that use Automation Services can help you implement views of your network and system resources from a service-driven operations perspective. You can group resources by business function (or service) and manage those services. You can monitor the services by using the line-by-line status monitor or the graphical monitor. Part II, *Service Management*, describes how you can set up these service-based views of your resources.

Automation Services also provides an event management facility that enables you to suppress unwanted messages and monitor messages in a multisystem environment. Part III, *Event Management*, describes how you can use this facility.

Note

Some products use neither the automation feature nor the event management facility. These products use Automation Services to provide only knowledge base management and presentation services.

Benefits

Automation Services provides the following benefits:

- Automation features that enable you to:
  - Improve the availability of services
  - Reduce the system startup and shutdown time
  - Reduce the recovery time for critical resources
  - Reduce operator intervention
  - Reduce the possibility of human errors
- Monitor features that provide visibility of the status of critical services and resources, and visibility of critical events.
- Event management feature that enables you to:
  - Process and respond to messages
  - Suppress unimportant messages
  - Monitor message flows to multiple consoles on a single screen, thus consolidating points of message management to a single point
  - Generate alerts to remind operators of significant events
- Event simulation feature that enables you to test your automation environment without affecting system operations. You can ensure that the system behaves as expected when you activate automation.
- Easy-to-use panels that enable rapid implementation of the automation environment.

About this Guide

The key to automation is to maximize the availability of services and resources to the business groups within the enterprise. This guide documents how to use the Automation Services functions to ensure the availability of information systems (IS) services required by the business groups.

This guide provides information that is common to the mainframe products that use Automation Services. To get started with the product you are using, read the user's guide for that product. This common user guide describes how to use the:

- Knowledge base, by defining and implementing your operations requirements
- Monitors to manage services, resources, and events

To perform the tasks described in this guide, the product region should already have been established.

What You Should Know Already

This guide is for operations analysts and operators working in the information systems (IS) group who need to automate and manage the processes of operating the networks or systems.

You should already be familiar with the operations environment, and with the service and resource availability requirements at your site.

Product Specific Information

This guide supports the mainframe products that use Automation Services. Where it is necessary to differentiate product specific information, the product is identified in parentheses as follows:

For …	The identifier is …
Unicenter NetMaster File Transfer Management (NetMaster for File Transfer)	FT.
Unicenter NetMaster Network Automation (NetMaster Automation)	NMA.
Unicenter NetMaster Network Management for SNA (NetMaster for SNA)	SNA.
Unicenter NetMaster Network Management for TCP/IP (NetMaster for TCP/IP)	IP.
Unicenter NetSpy Network Performance (NetSpy)	NS.
Unicenter SOLVE:Operations Automation (MSP)	MSP.
Unicenter SOLVE:Operations Automation (SOLVE:Operations Automation)	OS/390.
Unicenter SOLVE:Operations Automation (VOS3)	VOS3.
Unicenter SOLVE:Operations Automation for CICS (SOLVE:Operations Automation for CICS)	CICS.

2

Concepts

Automation Services is a service and message manager in that it enables you to:

- Ensure IS resource availability in the provision of services to the end user
- Manage messages, automate non-resource related operations, and consolidate the operator's view of messages

The product assists operations management by providing facilities to automate many aspects of the IS environment.

This chapter contains the following topics:

- Top-Down Approach to Operations
- Single System or Multisystem
- Services
- System Images and Versions
- Knowledge Base Components

Automation Services is then discussed in terms of its major component groups, to help you understand how it can be implemented to provide automation service.

Top-Down Approach to Operations

An operations environment can be viewed as comprising the following four levels (see Figure 2-1):

- The enterprise level consists of the major business functions that make up an organization (for example, distribution, marketing, and inventory control).
- The service level consists of applications (provided by the enterprise) that are grouped into business functions (for example, accounts payable and credit authorization). Applications consist of resources with defined relationships.
- The resource level consists of resources that are required to provide the services (for example, started tasks such as CICS and printers).
- The event level consists of events associated with the operation of the resources. Successful event management relies on the recognition of real and important events in the mass of messages and alerts flowing in a system. Besides messages, event management also handles timed events.

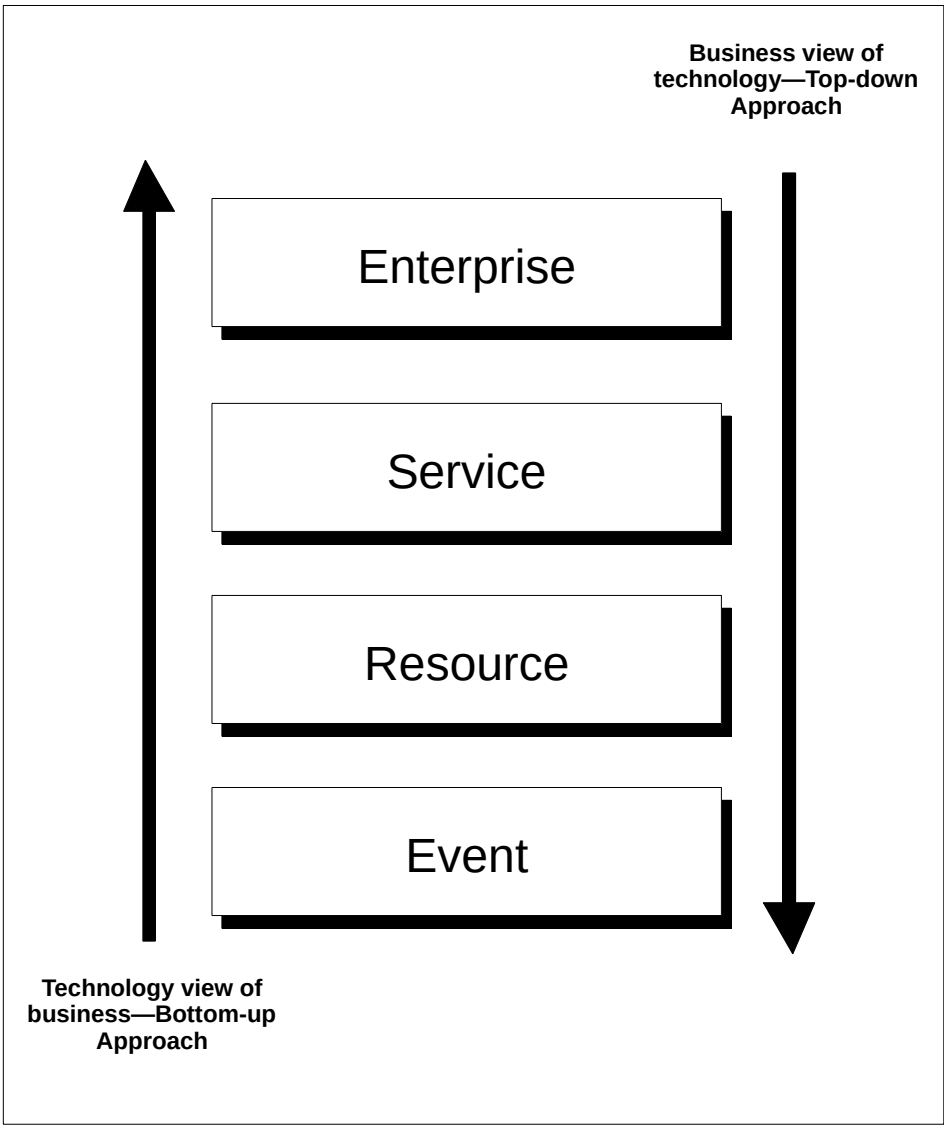
Historically, organizations have managed operations by monitoring and acting on events (bottom-up approach). You monitor the events and translate the events into resource information. Service requirements take a secondary role.

Automation Services, on the other hand, enables you to monitor the services and resources that are required for a business function directly (top-down approach). Service and resource availability or degradation is visible, and you can control the resources that make up a particular service to correct the interruptions to, or the degradation of, service provision.

Automation Services provides the following features for you to operate at the service, the resource, and the event levels of the model:

- The ServiceView feature of the product provides service level automation, monitoring, and control. See Part II, *Service Management*, for information about this feature.
- The ResourceView feature of the product provides resource level automation, monitoring, and control. See Part II, *Service Management*, for information about this feature.
- The EventView feature of the product provides event level automation, monitoring, and control. See Part III, *Event Management*, for information about this feature.

Figure 2-1. Operations Environment Model



Automation Services Approach

Automation Services approaches information systems operations by taking the top-down approach based on the concept of service management. A business view of IS technology needs to consider the enterprise (or organization) as a whole and then to break it down into the resources that affect business functions (services).

The information system is there to support and service the enterprise and its business functions. This means ensuring that, for example, all required resources are available to meet the current needs of a business function. It also means defining priorities for the business functions within the enterprise and then making sure that information system resources are adequate to meet those priorities.

Automating the control of resources is beneficial because it helps identify areas where money can be saved. Resource automation can help data center staff to understand the resources available to the organization and identify those that are critical.

The products that use Automation Services can automate a wide range of resources, including both software and hardware. They not only provide the mechanism to define the attributes of resources (including name, type, availability, starting method, stopping method, and other such control information), but also provide the ability to maintain the resources in user-specified desired states. Automation Services is aware of resources managed by the automatic restart manager (ARM) and will not try to restart those resources unless ARM has failed to restart them.

The service approach to automation introduces some important new features. Automation Services:

- Minimizes the time involved in event and resource management
- Enables operations to be monitored at the service level
- Enables operations staff to make decisions according to business priorities
- Assists in the recovery of services and resources, rather than just reacting to an event
- Introduces the notion that a service may require and exploit different levels of automation, taking account of cost benefits

The top-down approach streamlines and improves operator involvement, by introducing better automated procedures and computer-assisted manual procedures.

When this service-oriented approach is adopted, the automation product itself becomes one resource within the overall network and system management solution. The focus turns to the enterprise and the services it provides.

Single System or Multisystem

The region can run stand-alone, or can be linked to other regions, to provide a global status monitoring and control environment.

Automation Services provides focal point management to support multisystem operation (that is, management at a focal point with subordinates feeding information to it). An operator can monitor and control all the services and resources defined in the linked regions from a focal point region.

In the multisystem environment, the knowledge bases in the different regions are synchronized. You maintain the knowledge base in one region and the updates automatically propagate to the other linked regions.

Services

A service definition defines the availability of a service, and identifies the services and resources that are members of the service. Services can share members, and services can contain resources that are distributed across multiple systems. A service can be part of another service.

The region uses the defined availability of the service, the defined relationships between the resource members, and the defined operations policies and methods of the resources to maintain the availability of that service.

System Images and Versions

A system image represents a specific environment (for example, a development system, and all the resources and event rules required for it to function). You can define multiple versions of system images for your organization, and also specify information such as periods of resource availability.

Only one local system image can be active in a region at any one time. In a multisystem environment, each region must run a different system image. Operators can see the status of a resource defined to a system image by using Automation Services monitors. They can see the status of the resources on all connected systems from a single monitor.

You change the operational characteristics of the managed services and resources by loading a different system image. For example, you might have a production system image and a test system image, which are significantly different from each other, as well as a backup system image and a disaster recovery system image. The system images might contain different resources and activate different sets of event rules. Resource definitions are qualified by the system image name and version, and the resource name and class. Event rules are contained in the ruleset that is associated with the system image.

Version Control

You can copy a system image and give the copy a new name and version number. This approach is valuable for change management. It gives you the option to start up one system image, knowing that if it fails (for example, if resources are not correctly defined or there are bugs in a new application), you can go back to the previous system image that you know works. Usually, you define three system images—current, old, and new, but you can keep up to 9999 versions of a system image.

Knowledge Base Components

The Automation Services knowledge base contains the following groups of components:

- System images—each system image has its own resource definitions, resource availability maps, and processes. A system image can also activate a set of event rules.

  The special $SERVICE image contains service definitions. Version 0001 of this image is always active in addition to the normal active system image.

- Often-used definitions—global processes, resource groups, icons, icon panels, status monitor filters, and message profiles are subject to ongoing change. New ones may be added as the networks and systems change.

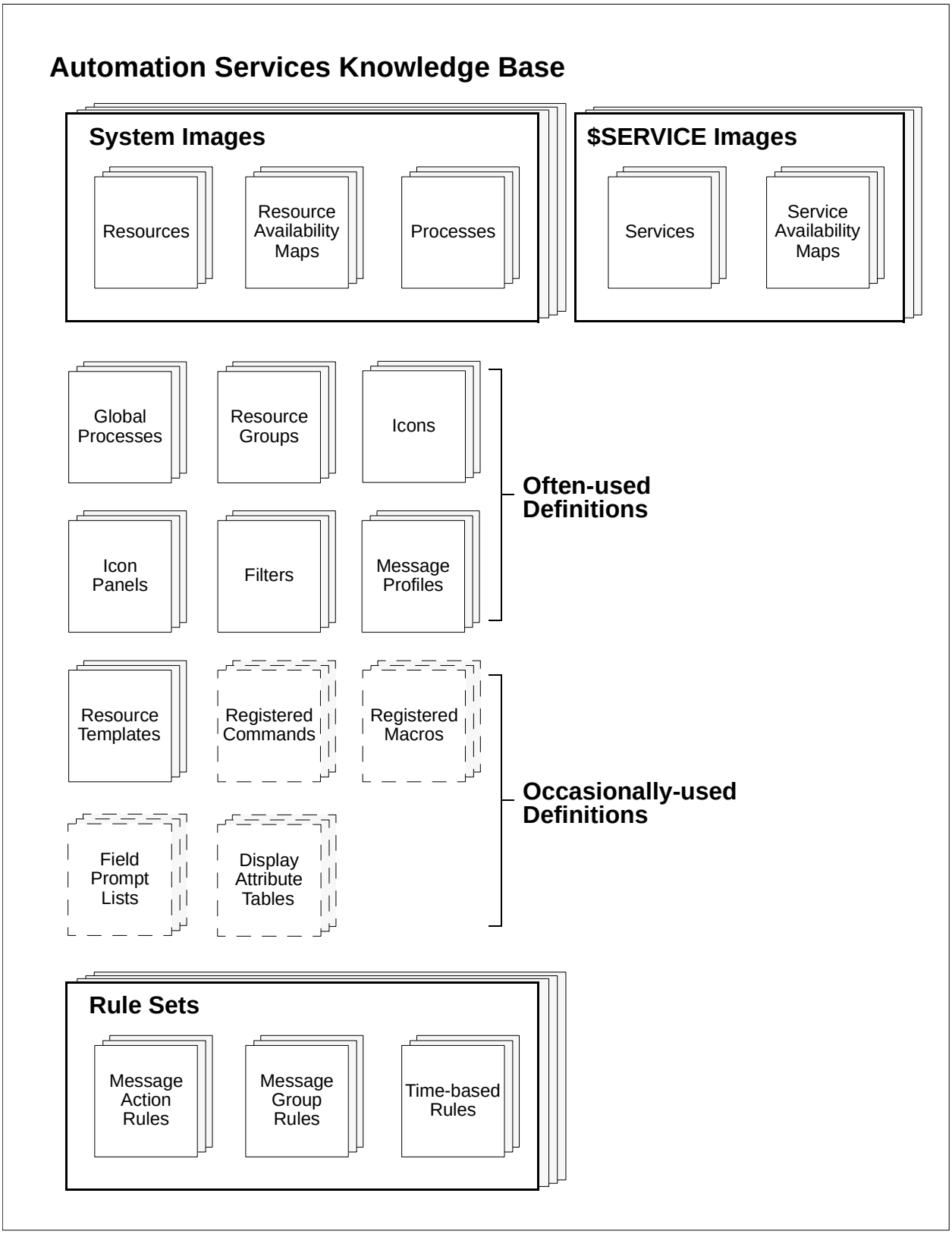
- Occasionally-used definitions—resource templates, registered commands and macros, field prompt lists, and display attribute tables are generally defined or customized when the product is first installed. Once defined, they rarely require modification.

- Rulesets and event rules—a ruleset is a collection of event rules. Each ruleset can contain message action rules, message group rules, and time-based rules.

You define images for the different systems in the organization. You define filters and icons to group the resources defined in the system images to provide specific views of the resources for the different business functions. You define event rules to automate non-resource specific tasks.

Figure 2-2 shows these groups within the knowledge base. The boxes that do not have a solid outline identify components that are described in the *Automation Services Administrator Guide*.

Figure 2-2. Automation Services Knowledge Base Components



Part II

Service Management

3

About Service Management

Automation Services gives you a business view of the services you are providing. To do this, you need to define and group resources into the provided services.

This chapter contains the following topics:

- Managing Services
- Features
- Implementing the Operations Environment
- Developing System Images That Contain the Resources You Want to Manage
- Defining Services
- Specifying the Availability of Services and Their Members
- Monitoring Service-driven Operations

For a tutorial about how to define a service, see the *Automation Services, Setting Up Service Definitions, A Tutorial* manual.

Managing Services

You can obtain a business view of the services you are providing by grouping your resources by business functions. You can then monitor these business functions. If an operational problem occurs, you can list the resources to determine which resources are causing the problem.

Where your service uses resources from multiple systems, you can include the required resources on the different systems within a single service.

Automation Services is used by products that support different classes of resources (for example, NetMaster Automation supports SNA resources and SOLVE:Operations Automation supports OS/390 resources). If you have the relevant products, you can build a service that includes resources supported by those products.

Features

Automation Services offers the following features that help you manage your networks and systems from a service-driven operations perspective:

- ResourceView enables you to define resources, and specify policies and methods to manage their operations.
- ServiceView enables you to implement the management of the defined resources in terms of business functions or services.
- Operation states help you determine the status of managed services and resources.
- Operation modes influence how the defined services and resources are managed.
- Monitors display managed services and resources textually (line by line) or graphically (as icons).
- Availability maps schedule the operations requirements of the managed services and resources.
- Process definitions enable you to implement complex operations methods.
- Reporting enables you to obtain operations statistics about the managed services and resources.
- Knowledge base provides a repository for storing operations policies and methods.
- Multisystem support enables you to monitor and operate your networks and systems from a focal point region.

AutoAssist

AutoAssist enables you to set up a region and define the operations policies of your organization to the knowledge base quickly. The feature provides the following facilities:

- Assisted first-time region startup.
- Depending on the product you are using, there are facilities that discover the resources on your system and define them to the knowledge base. For example, the SOLVE:Operations Automation product provides an Express Setup Facility, an Auto Populate Facility, and assisted resource learning.
- Templates that enable you to create definitions of a specific type of resource quickly.
- Prompted fields to assist you in the data entry process. A prompted field provides a list of valid or suggested values from which you select the appropriate one.

ResourceView

ResourceView enables you to define the operations policies for the resources required by your organization. You can specify the following characteristics of a resource:

- Over what periods of time the resource must be available
- How the resource is started
- How the resource is stopped
- How the resource state is checked
- How the region responds to messages associated with a resource

You define resources to system images. Each image represents a view of the managed resources on a single system in the organization. When the regions start up, these images are loaded on the relevant systems. Each region manages the resources local to the system.

Automation Engine

The automation engine implements your organizations operations policies by reacting to events, and correlating resources with their availability schedules. This engine automatically reconciles the *actual state* of a resource with its *desired state*. This engine implements the policies of an organization as defined in the knowledge base, by reacting to events automatically.

Desired State Management

The region monitors the actual and the desired states of the resources that make up the business functions (services), and attempts to ensure that the actual states match the desired states. You define criteria in the knowledge base that enable the region to maintain the availability of resources. These criteria include:

- Availability based on:
  - Time schedules
  - Relationships to other resources
  - External events such as the starting and stopping of batch jobs
- The ability to temporarily override predefined operational states

Startup, Shutdown, and Recovery

The region can automate the startup and shutdown requirements of defined resources:

- At the system level, the automation engine of the region initiates startup and shutdown on the basis of parent and child relationships between the resources defined in the knowledge base.
- At the resource level, the automation engine starts, stops, and recovers individual resources, depending on the parent and child relationships defined between them. The relationships ensure the correct starting or stopping sequence.

  The region monitors the status of the resources continuously. If the actual state of a resource deviates from that of the desired state, the automation engine acts on it to bring the actual state back to what is desired. If the parent in a relationship fails, the engine not only recovers the parent but also ensures the recovery of any failed children.

The automation engine starts and stops resources with or without operator intervention, as specified in the definitions.

If a failed resource *cannot* be recovered, it is highlighted on the monitor and the operator can take appropriate action. An exit to a problem management application can be specified to raise a problem ticket automatically.

ServiceView

ServiceView enables you to define the operations policies for the services required by your organization. By using service definitions, you can specify the following characteristics of a service:

- Over what periods of time the service must be available.
- The services and resources (the members) that are required for the provision of the service. Services can share members and can include resources from different systems.
- How a member status affects the provision of the service.

Active service definitions in the knowledge base determine the availability of services.

You define the availability of a service by specifying the desired state of the service at particular times. In a multisystem environment, you can specify a system as the *service automation focal point system*, and the scheduled times refer to the local times of that system.

The region uses the defined availability of the service, the defined relationships between the members, and the defined operations policies and methods of the members to maintain the availability of that service.

When a service starts, it requires that all its members be also started, irrespective of their desired states. When a service stops, it removes its availability requirements on its members.

Operation States

The following states indicate the status of a service or resource defined in an active system image:

- Desired state is the required state, as specified, for example, in the availability map. The desired state can be either ACTIVE or INACTIVE.
- Actual state is the state determined during operation (for example, ACTIVE). The states are determined by the rules specified in the individual definitions.
- Logical state is user-modifiable and indicates how well the actual state is tracking the desired state. For example, an INACTIVE actual state and an INACTIVE desired state can be indicated by an OK logical state, while a STOPPING actual state and an ACTIVE desired state can be indicated by an INERROR logical state.

The logical state determines the display attributes. For example, the default status display of an INERROR state is high intensity white text in reverse video.

Authorized users can customize the attributes of the logical states. See the *Automation Services Administrator Guide* for information about customizing the logical states.

Desired States

The desired states are as follows:

- ACTIVE indicates that the monitored item should be available for use.
- INACTIVE indicates that the monitored item is *not* required.

Actual States

The actual states are as follows:

- ACTIVE indicates that the monitored item is available for use.
- DEGRADED indicates that the monitored item is available for use but is *not* operating in the optimal condition.
- FAILED indicates that the monitored item has a problem and requires manual intervention.
- INACTIVE indicates that the monitored item is unavailable for use.
- STARTING indicates that the monitored item is starting.
- STOPPING indicates that the monitored item is stopping.
- UNKNOWN indicates that the region is unable to determine the actual state of the monitored item.

Logical States

The default logical states are as follows, with the worst state listed first:

Note

You can change the default logical state settings. Enter the **/ASADMIN.A** path to access the attribute tables.

- UNKNOWN indicates that the region is unable to determine the actual state of the monitored item.
- INERROR indicates that an error condition has occurred. Normally, the condition means that the region is unable to execute the required operations method for the monitored item successfully. The region releases control of the item. You should investigate and correct the problem, then restart automation for the item.
- FAILED indicates that the monitored item, which has a desired state of ACTIVE, failed to start. You should investigate and correct the problem, then restart automation for the item.
- ATTENTION indicates that the monitored item requires attention. For example, if the actual state is INACTIVE while the desired state is ACTIVE, but the operation mode is MANUAL, the region has *no* control over the item and thus cannot bring the item to the desired state. The ATTENTION logical state indicates that you need to intervene to bring the item to the desired state.
- DEGRADED indicates that the monitored item, which has a desired state of ACTIVE, has degraded. If the region is unable to change the logical state to OK, you might need to investigate the cause of the degradation and correct the problem.
- PENDING indicates that the region is waiting for certain conditions to be satisfied before bringing the monitored item to the desired state. For example, during starting, a child needs to wait for the parents to become active first.
- STARTING indicates that the monitored item is starting.
- STOPPING indicates that the monitored item is stopping.
- OK indicates that the actual state is as desired.

Operation Modes

When you define a service or resource, you specify its operation mode. The operation mode determines how the region acts on that service or resource. Table 3-1 describes the modes.

Table 3-1. Operation Modes

Operation Mode	Description
AUTOMATED	The region monitors, and has full control of, the monitored item.
IGNORED	This mode is the same as the MANUAL mode except that the logical state of the monitored item is always displayed as OK. Typically, this mode is applied at times when the item requires no attention.
MANUAL	The region monitors, but has no control over, the monitored item.
OFF	The region neither monitors, nor controls, the item.

During operation, you can override the mode by using commands.

Global Operation Mode

The global operation mode is the highest mode that any resource can run in. You can apply a MANUAL global operation mode to a system image to force all its members to operate in the MANUAL operation mode. For example, if the global operation mode is MANUAL and the operation mode of a resource is AUTOMATED, the resource can only run in the MANUAL operation mode, unless the global operation mode is changed to AUTOMATED.

Typically, you use the MANUAL global operation mode when you load a new system image. You can then monitor its behavior for a while before switching to automated operation.

You set the initial global operation mode of the loaded system image in the AUTOIDS parameter group. To access the list of parameter groups, enter the **/ICS** shortcut.

Monitor Features

The region has two, color-coded monitor displays:

- A status monitor, which displays status information textually, line by line
- A graphical monitor, which displays status information graphically, by means of colored *traffic light* icons

Both monitors are real-time and information is dynamically updated. Changes in the monitored status are therefore reflected by your monitor as they occur.

The monitors can be customized to display status information selectively. You define the customization requirements in the knowledge base as status monitor filters and graphical monitor icon panels.

You can use these definitions to set up multiple status and graphical monitors, each providing a different view of the monitored systems networks.

Generic Commands

The region provides common actions (such as startup and shutdown) that are applicable to different classes of resources, and insulates you from the specific implementation of these types of actions. That is, you can use a single command to perform a similar function on different classes of resources. Depending on the resource, the command behaves differently. For example, you can use the same command to start a printer or a tape unit.

Availability Maps

You can define the availability requirements for a service or resource by using calendar-based availability maps.

Processes

You define processes to perform complex processing (for example, a complex resource starting operation). A process is a series of processing steps that can be executed singly or in sequence. Each step executes a macro, which is an NCL procedure. The product comes with a set of macros, but you can implement your own macros if the supplied set does not meet your requirements.

Each macro provides a panel-based parameter entry interface to simplify the implementation of a process.

A process can be one of the following types:

- A global process that is available for use at all times
- A system image-specific local process that is available for use only if that particular image is active on the local system

Reporting

You can generate reports about the following:

- Statistics about the amount of automation being performed (for example, messages suppressed, messages processed, and analysis of message traffic).
- Service and resource availability reporting, to indicate to management whether service level availability objectives are being met. These reports show statistics about required start and stop information, as follows:
  - The actual time a monitored item was available, and the percentage of time for which it was available
  - The number of times a monitored item became available, and the number of times it became unavailable
  - The time a monitored item was unavailable
- Knowledge base reporting (see Appendix I, *Reporting*).

The region also writes system management facility (SMF) records (on MSP and OS/390 systems) or System Management Service (SMS) records (on VOS3 systems) containing EventView, ResourceView, and ServiceView statistics. A product such as SAS can read these records and produce reports. For information about EventView, see Part III, *Event Management*.

The *Automation Services Administrator Guide* describes the structure of these records.

Knowledge Base

The knowledge base provides a consistent but flexible format for defining operations policies and methods as they relate to the availability of managed services and resources. Sample templates, which contain predefined operations methods, help you define common resources. You can define your own templates. The knowledge base provides the following features:

- Support for multiple versions of managed systems. The versions are defined as versions of system images. Each image version is the equivalent of a single operating environment (for example, a production system image or an image for handling system disaster recovery). A image contains all the resource definitions and event rules required to automate the operation of that system.
- Menu-driven panels enable you to define information to the knowledge base, including:
  - Resource definitions
  - Service definitions
  - Availability maps
  - Processes
  - Relationships that define the parent-child dependencies of resources for startup, shutdown, recovery, and monitoring
  - Monitor customization filters and panels
  - Consolidated console message profiles (see Part III, *Event Management*)
  - Event-based automation rules (see Part III, *Event Management*)

Multisystem Support

The region provides multisystem support by using INMC links.

Peer-based Automation

Automation Services provides focal point management to support multisystem operation (that is, management at a focal point with subordinates feeding information to it) as follows:

- Peer-to-peer architecture—supports full connectivity between multiple regions. Regions linked in this way are known as focal point regions. (A stand-alone region is also regarded as a focal point region.)

  When regions are communicating with each other, authorized users can monitor and control all managed resources from any terminal connected to any focal point region.

- Subordinate—enables you to reduce the amount of traffic in your multisystem environment. You link subordinates to focal point regions that provide central monitoring and control. A subordinate has visibility and control of the locally managed resources only. You cannot monitor a service in a subordinate region.

In a multisystem environment, each region can run independently of the other regions. If no communication links are available, each region still provides full monitoring, control, and automation of its own managed resources.

Peer-based automation means that regions run independently of each other in a multisystem environment.

Multisystem Knowledge Base

You can update the knowledge base from any region. The knowledge bases in connected regions are synchronized and updated (where appropriate) to reflect any new information, with automatic recovery in the event of link failure.

Note

For linked *focal point regions*, synchronization is complete and the focal point knowledge bases duplicate each other.

For linked *subordinate regions*, synchronization is complete only to the extent of the relevant definitions in the knowledge base. For example, a subordinate knowledge base does not contain all system images. A subordinate knowledge base contains only those images that represent the environment the subordinate is managing.

Management of Services

A service can contain resources that are distributed across systems. Operators can monitor and control these resources from any connected focal point region. Also, operators can monitor alerts and messages from any connected focal point region.

The region on each system ensures the availability of the resources on that system according to the service requirements.

Management Services

Management Services provides a central core of basic functions and services to various products, including those products supported by Automation Services.

Management Services supports systems operations and includes the following services:

- Activity logs—these logs record the activities that occur within a region. You can search through a log by time, date, or a text string.
- Messages and codes database—this database provides help information on error messages and codes (for example, VSAM OPEN macro return codes).
- Print Services Manager (PSM)—PSM enables you to control the physical printing of reports on JES or network printers. You can view a report before printing it.
- Network Control Language (NCL)—this language enables you to develop procedures for performing special tasks. Automation Services commands and macros are implemented in NCL.

Implementing the Operations Environment

You define resources to the Automation Services knowledge base to provide the following:

- Management of desired state of resources
- A service-driven operations perspective of your networks and systems

Successful implementation of the operations environment requires an understanding of the operations requirements. This section outlines the steps you can follow to perform the implementation, starting with monitoring and ending with automation.

Managing the Desired State of Resources

You might want a resource to be active over certain periods of time and to be inactive over other periods of time. You might also want to include the resource in a service.

If you want to do the above, you must define the affected resources to the knowledge base. You define the resources by first defining a system image and then by defining the resources to that image. You specify the availability requirements of these resources.

Providing a Service-driven Operations Perspective of Your Networks and Systems

To put your operations in a service-driven perspective, you group related resources in services. You specify the availability requirements of those services. The services in turn ensure the availability of its members.

Defined services are visible on the status monitor in a focal point region. You can tell whether a service is healthy at a glance.

Multisystem Consideration

In a multisystem environment, ensure that automation is working in the first region before connecting the next region, progressing step by step to build up the environment. See the *Automation Services Administrator Guide* for information about how to connect regions.

Typical Implementation Tasks

To implement Automation Services to manage your business functions and the resources that support these functions, perform the following suggested sequence of tasks:

1. Implement Automation Services to monitor resources.
2. Implement Automation Services to automate operations.
3. Implement Automation Services to manage business functions.

Note

The steps described in the following sections require you to control the system image and use the status monitor. See Chapter 13, *Monitoring and Controlling Services and Resources*, and Chapter 14, *Controlling the System Image*, for information about those tasks.

Task 1—Implement Automation Services to Monitor Resources

A typical sequence of steps is as follows:

Step 1. Review your operations requirements. Determine the business functions and resources to be managed.

Step 2. Create a system image and add resource definitions to it. Your product might provide an AutoAssist facility that helps you do this.

Step 3. From the status monitor, use the LOAD command to load the system image with a MANUAL global operation mode. The mode allows monitoring and operator control of the resources, but disables automation.

To access the status monitor, enter the **/RMON** shortcut.

Key Concept

After you loaded an image, you can define additional resources to it and the defined resources will immediately become available to the region.

Step 4. Use the status monitor to learn about the behavior of the resources during the course of operation. Define status monitor filters to customize your view of the monitor.

You can define a status monitor filter from the Status Monitor Filter List panel. To accessed the list, enter the **/ASADMIN.F** path.

Step 5. Generate icon panels that enable you to use the graphical monitor, and use the graphical monitor to provide you with a different view of the monitored resources.

The graphical monitor Administration Menu provides options that enable you to implement icon panels. To access the menu, enter the **/GADMIN** shortcut. To access the graphical monitor, enter the **/GMON** shortcut.

Step 6. When you are satisfied with the monitoring setup on one system, connect another system into the Automation Services environment. Repeat Step 1 through Step 5 for this new system.

See the *Automation Services Administrator Guide* for information about how to connect and synchronize the Automation Services knowledge base.

Task 2—Implement Automation Services to Automate Operations

A typical sequence of steps is as follows:

Step 1. Review your operations to determine the availability requirements of and the relationships between the defined resources on one system.

Step 2. Use the event simulator to check the expected responses to events from the resource definitions (see Chapter 10, *Testing Definitions by Using Simulated Events*). Refine the resource definitions and relationships in the system image accordingly.

To access the event simulator, enter the **/EADMIN.E** path. To access a resource definition from the status monitor, enter **DB** beside the definition.

Step 3. When you have learnt enough about the resource behavior and have refined the operations requirements in the resource definitions, enter **MM** (Mode MANUAL) beside all resources, then use the GLOBAL command to change the global operation mode to AUTOMATED to enable automation. Because of the application of the MM command, the resources remain in the MANUAL operation mode.

Step 4. Bring the resources, a small number at a time, in a logical manner, into the AUTOMATED operation mode by entering MR beside them. By using the monitor, ensure that automation behaves as required. Refine the system image to correct any variations from the required behavior.

Step 5. Continue bringing resources into the AUTOMATED mode until the automation requirements are satisfied for this system image.

Step 6. Use the EventView message learning facility to identify any non-resource related messages for which rules need to be created (see Chapter 19, *Message Learning*). Create those rules. You can suppress resource-related messages. Suppressing resource-related messages from EventView does not affect resource automation.

To access the Event Administration menu, enter the **/EADMIN** shortcut.

Step 7. Create timer rules to schedule non-resource related actions.

Step 8. Associate the event rules (through a rule set) with the active system image to load the rules. Use the message monitor (Operator Console Services (OCS)) to check the operation of the message rules and timers. Use the event simulator to check the expected responses to events from the message rules.

To access the message monitor, enter the **=O** path.

Step 9. When you are satisfied with the automation on one system, repeat Step 1 through Step 8 for the next connected system.

Task 3—Implement Automation Services to Manage Business Functions

A typical sequence of steps is as follows:

Step 1. Review your operations to determine the business functions to be managed.

Step 2. In a focal point region, define services to represent these business functions and monitor their behavior. Refine the services to correct any variations from the required behavior.

You can define a service from the Service List panel. To access the list, enter the **/SADMIN.S** path.

Step 3. Create or refine any services that require resources from multiple systems.

Developing System Images That Contain the Resources You Want to Manage

After a region is established, you can create the resource definitions that are the building blocks for your service-driven operations environment. For each system, you develop a system image that contains these definitions. This section contains information about the main procedures involved in developing an image of the resources you want to manage.

Implementing a System Image

You use system images as the basis for managing the resources in your organization. An image represents a view of the resources on a particular system in your organization, and contains the operations policies and methods associated with those resources. You can associate an EventView rule set with an image to manage non-resource related events. Define an image for each of the systems managed by a region.

Typical stages in implementing a system image are discussed below and shown in Figure 3-1.

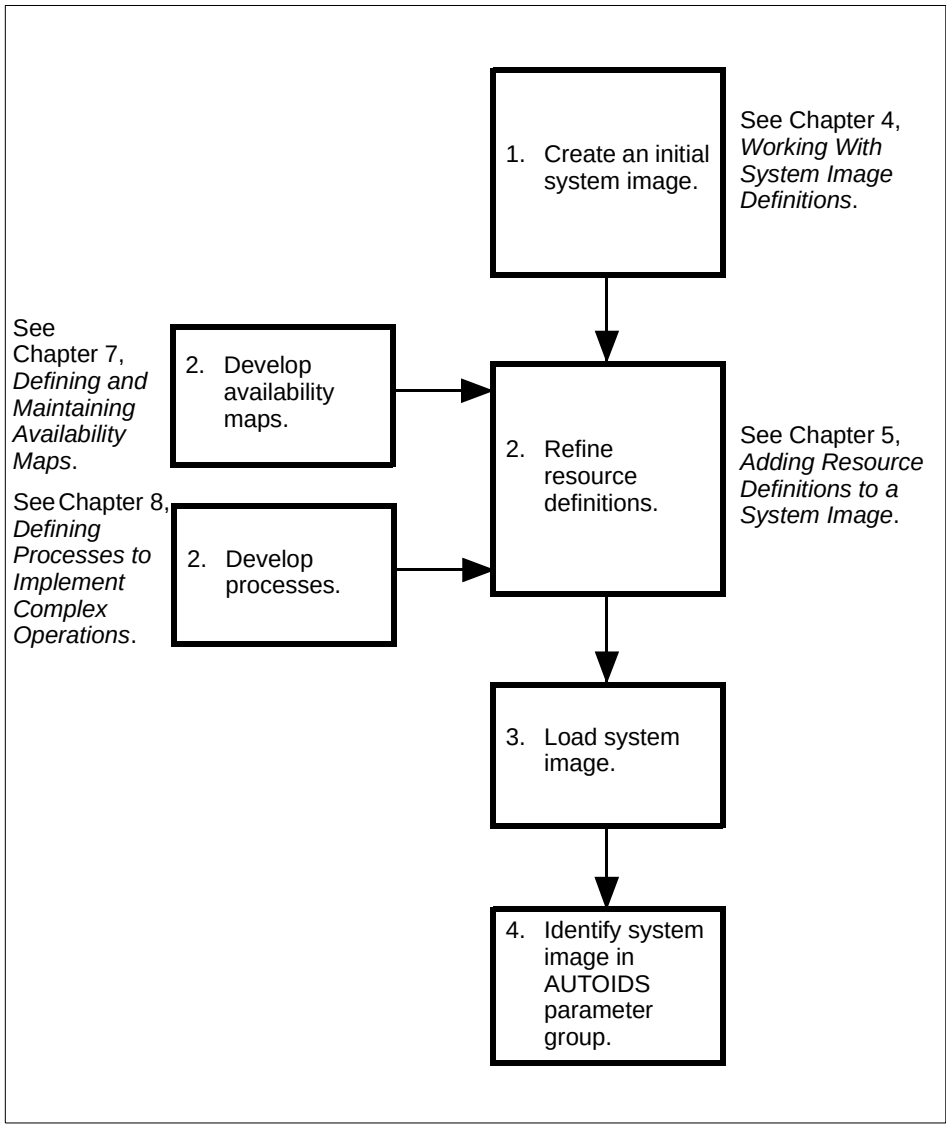
Step 1. Create an image for your system and add definitions of the resources you want to manage to it.

Step 2. Refine the resource definitions to suit your requirements. Develop availability maps to specify the availability of the different resources. Develop processes that automate tasks according to resource activities.

Step 3. Load the system image. If the operation of the defined resources is not satisfactory, further refine the definitions; if the operation of the defined resources is satisfactory, proceed to the next step.

Step 4. Identify the working system image to the AUTOIDS region parameter group. This results in the image being loaded every time the region starts up.

Figure 3-1. Implementing a System Image



System Image Components

A system image uses the following components to define the operations environment:

- System image definition—a system image definition reflects the operations environment of a single system. Each image may have several versions in the knowledge base. You can modify image definitions and add new definitions if required.

  Chapter 4, *Working With System Image Definitions*, describes how to create system image definitions.

- Resource definitions—after you have created a system image definition, you can define the resources to the image. Each image can contain several classes of resource definitions, as supported by the products you are using (for example, started tasks, printers, and SNA groups), which the region controls to whatever degree you specify. You can modify existing definitions and add new definitions if required.

  When you have defined the resources in an image, you can define the relationships between them. The relationships indicate the dependencies between the resources, and determine the order in which resources are acted upon (for example, during a shutdown sequence).

  Chapter 5, *Adding Resource Definitions to a System Image*, describe how to define resources to an image. See your product user's guides for product specific information.

- Availability maps—availability maps determine changes to the default availability of resources. A map can also enable processes to start at defined times. A map can be used by more than one resource.

  Chapter 7, *Defining and Maintaining Availability Maps*, describes how to define availability maps.

- Processes—processes automate the processing of multiple commands (for example, the complex shutdown of a database region). A process can be used by more than one resource in a system image.

  Chapter 8, *Defining Processes to Implement Complex Operations*, describes how to define processes.

Defining Services

When you have populated the system images with resources, you can define the services provided by them. A service comprises resources and other services as members. A service can include resources that reside on other systems.

You specify how the status of a member affects the provision of the service. The failure of some members can stop a service, while the failure of other members might only degrade the service.

You specify the availability of the service by using an availability map. The availability is defined in terms of the time on the *service automation focal point system*. In a multisystem environment, you specify a system as the focal point system, and the scheduled times refer to the local times for that system. If the map schedules the starting of processes, the processes are started in the region on that system only.

Chapter 6, *Defining and Maintaining Services*, describes how to define services.

Specifying the Availability of Services and Their Members

The availability of a service overrides the availability of its members. If a member always operates as part of a service, you can let the service handle the availability of that member.

To let the service control the availability of a member, set the desired state of the member to be always INACTIVE by using one of the following methods:

- If the default desired state is ACTIVE, attach the member to an availability map that specifies an INACTIVE desired state for all times.
- If the default desired state is INACTIVE, you do not need to attach an availability map to the member. The member is desired inactive by default.

Note

The default desired state is set in the AUTOIDS parameter group during region initialization. Once set, the default should not be changed, as the availability requirements defined in the knowledge base are based on this default. The Customization Parameters panel lists the region parameter groups. Enter the **/ICS** shortcut to access the panel.

Activation of a Service

When the service is required to become active, it desires the member to become active, overriding the desired state set by the availability map or by default.

When the desired state of a service changes from INACTIVE to ACTIVE, all its members need to be started. The region places an ACTIVE desired state override on each of the service members. If any of the members are inactive, the region starts those members. The service is indicated as active when a predefined percentage of its members are active.

Inactivation of a Service

When the desired state of a service changes from ACTIVE to INACTIVE, this service is indicated as inactive immediately. The region removes the ACTIVE desired state overrides from the service members. The members might or might not become inactive, depending on, for example, whether they are required by other services.

Status of a Service

The status of a service with a desired state of ACTIVE is determined by the status of its members. The failure of some members can stop a service, while the failure of other members might only degrade the service. You can distinguish between a failed service and a degraded service and respond accordingly. You specify the importance of a member to the service in the service definition.

Monitoring Service-driven Operations

A region provides two monitors that you can use to monitor the status of services and resources: the status monitor and the graphical monitor. You can define status monitor filters and graphical monitor panels to customize the view of the monitored environment.

Status Monitor Filter

A status monitor filter enables you to specify what are displayed on the status monitor. For example, a filter is provided that limits the display to the status of services. When the status of a service indicates a problem, you can zoom in on the service to view the status of the underlying resources.

Predefined filters provide the following versions of the status monitor:

- Service monitor (available in focal point regions only)
- Resource monitor

Graphical Monitor Icon Panel

A graphical monitor icon panel enables you to design an iconic representation of the monitored services and resources. The display attributes of an icon reflect the status of the represented member that is in the worst state. You can zoom in on the members of an icon.

4

Working With System Image Definitions

You define the operations requirements of the resources to be managed on a system in a system image. A system image consists of the image definition, and a number of resource definitions, availability maps, and processes. An image can also activate a set of event rules. The region uses the information in the local active system image to monitor and control the resources on the local system.

This chapter contains the following topics:

- When Do You Need to Define System Images?
- Implementing System Images
- Accessing System Image Definitions
- Actions That You Can Perform on a System Image
- Defining a System Image
- Maintaining System Images

Chapter 5, *Adding Resource Definitions to a System Image*, describes how to define the operations requirements of resources to an image. See your product user's guides for any supplied AutoAssist facilities that can help you create your initial system image.

When Do You Need to Define System Images?

You must create a system image definition before you can define the resources you want to manage.

Implementing System Images

Note

The $TEMPLAT system image name is reserved for templates. Chapter 9, *Defining and Maintaining Resource Templates*, describes how to work with templates.

Define a system image by giving it a name and assigning a version number. You must also nominate a home system on which the image can be loaded.

Note

If you are defining a system image for a subordinate region in a multisystem environment, the image name is restricted to that specified during the linking operation (see the *Automation Services Administrator Guide*).

You assign different version numbers to create different views of the managed resources in the knowledge base. For example, the current live version of an image is version 0001, the new version (which contains changes that are not yet implemented) is version 0002, and the old version (a view of the image as it was before the changes in the current version were implemented) is version 0003.

You can also use the version to reflect the creation date of the system image (for example, 9805 for May 1998).

The following two examples show you how you can set up the different image versions. Example 1 shows the use of three images, for old, current, and new versions of an image. Example 2 shows the use of up to 9999 versions of resource management variants on a single system.

Example 1

You are responsible for the IS department of the eastern branch of a company, and you have defined a development system image called EASTTEST 0001. When the branch needs to add or change some definitions, you copy EASTTEST 0001 to a new image called EASTTEST 0002. You change the contents of EASTTEST 0002 and test it until it is working properly.

When the changes are working, you copy EASTTEST 0001 to a new image called EASTTEST 0003. This new image can be used as a backup if EASTTEST 0002 fails for any reason. You then load EASTTEST 0002 to make it temporarily the current image and delete EASTTEST 0001. You can then copy EASTTEST 0002 to a new EASTTEST 0001 and load this EASTTEST 0001 image.

The next time you want to update the current system, you update EASTTEST 0002, test it and back up EASTTEST 0001 to EASTTEST 0003. You then proceed as before to create a new EASTTEST 0001 by copying the tested EASTTEST 0002.

Example 2

This example assumes that there is enough DASD space to store the 9999 images of a system.

You can have 9999 images of a system. Therefore, it is possible to create a new image each time you update a system. The first version might be called EASTTEST 0001, the next update might be called EASTTEST 0002, and so on, until EASTTEST 9999 is reached. The next update would then be called EASTTEST 0001 again. This method enables a data center to keep extensive backups.

Accessing System Image Definitions

Enter the **/RADMIN.I** path to access the list of system image definitions stored in the knowledge base.

Active system images have the following color coding:

- White is used to indicate the active system image used by the local region.
- Turquoise is used to indicate the active system images used by connected regions.

Updating an *active* system image has an immediate effect on the operations of the resources controlled by the region.

Actions That You Can Perform on a System Image

You can perform the following actions from the System Image List panel:

F4 (Add)	Displays the System Image Definition panel for you to add a system image definition. Figure 4-1 on page 4-5 shows a sample panel.
S, /, or B (Browse)	Displays a system image definition in BROWSE (read-only) mode. The definition includes the system name and version, short and long descriptions of the image, and system-generated statistics on when the image was created and last updated. If you are authorized to update image definitions, you can press F4 (Edit) to switch to the UPDATE mode.
U (Update)	Displays a system image definition in the UPDATE mode.
C (Copy)	Displays a system image definition in the COPY mode. You must change at least the image name or the version, depending on the reason for the copy. For example, if you are defining a new production system that has resources and automation requirements similar to a system you have already defined, you would give the copied image a new name, such as EASTPRD1. When you apply this action, all resource definitions, availability maps, and processes in the system image are also copied. You can use this action code to merge two system images (see the section, *Merging System Images*, on page 4-6).
D (Delete)	Displays a deletion confirmation message. Press ENTER to delete the system image and all its components. Press F12 (Cancel) to retain the system image.
R (Resources)	Displays the Resources List panel that lists the resource definitions in the selected system image. From the Resources List panel, you can access the resource definitions.
GRT (Graphic Rsc Tree)	Displays graphically all resource relationships in the selected system image.
L (Load)	Enables you to load a system image. For more information about the loading operation, see the section, *Loading a System Image*, on page 14-3.

Defining a System Image

To add a system image definition, press F4 (Add) from the System Image List panel. A System Image Definition panel is displayed (Figure 4-1 on page 4-5 is an example). You can now define the system image.

Example

In this example, you are adding a system image definition to the knowledge base. The system image is called EASTTEST, which represents a development system of the same name (the home system). The *Automation Services Administrator Guide* contains more details about system images in a multisystem environment. Figure 4-1 shows the completed definition.

Figure 4-1. Example—System Image Definition Panel

```
SOLVPROD----------- ResourceView : System Image Definition -------------------
Command ===>                                                   Function=ADD


System Name ......... EASTTEST
Database Version .... 0001
Home System ......... EASTTEST  (...where this Image will load)

Short Description ... Development system for Eastern
Long Description .... Handles application development and testing by Eastern
                      branch programmers.  Contains all the resources
                      necessary to maintain this environment.
                      ____________________________________________________

EventView Ruleset to Activate + ________



 F1=Help      F2=Split     F3=File      F4=Save
                           F9=Swap                              F12=Cancel
```

For products that support Automation Services event management, you can associate an EventView rule set with the system image by using the EventView Ruleset to Activate field. The rule set is activated when the system image becomes active, thus enabling event-based automation through the active event rules.

Note

By default, ResourceView and EventView rule actions are not executed in the MANUAL operation mode. The actions, however, can be enabled by using the Perform Action in Manual Mode? field of the AUTOIDS region parameter group.

Maintaining System Images

You can browse and update system image definitions, and copy and delete system images, from the System Image List panel.

Note

If you make changes to the definitions belonging to an active system image, the changes become effective in the active image immediately.

In a multisystem environment, this active image can be in one of the connected regions.

Merging System Images

You can use the C (Copy) action code to merge two system images. When you merge the images, you can specify whether you want to overlay existing definitions. The target system image contains the merged definitions.

Merge a source system image into a target system image as follows:

Step 1. On the System Image List panel, enter **C** beside the source system image. The system image definition is displayed.

Step 2. Identify the target system image in the System Name and Database Version fields. Press F3 (File) or F4 (Save) to initiate the merging operation. You are prompted to confirm the operation.

Step 3. Indicate whether you want to overlay definitions that already exist in the target. Press F6 (Confirm) to start the actual merging operation. A panel is displayed to indicate the progress of the operation.

Figure 4-2 shows an example that overlays the EASTTEST 0001 system image onto the EASTPRD1 0001 system image.

Figure 4-2. Merging System Images

```
SOLVPROD------------- ResourceView : System Image List ------------------------
Command ===>                                                   Scroll ===> CSR

    S/B=Browse U=Update C=Copy D=Delete R=Resources GRT=Graphic Rsc Tree L=Load
    Sysname   Ver    Short Description
    EASTPRD1  0001   Eastern branch production 1
C   EASTTEST  0001   Eastern branch development
```

```
SOLVPROD------------- ResourceView : System Image Definition ------------------
Command ===>                                                    Function=COPY


System Name ......... EASTTEST
Database Version .... 0001
Home System ......... EASTTEST
```

Change values to identify target, then press F3.

```
SOLVPROD----- Automation Services : Confirm System Image Merge  ---------------


                            MERGE TWO SYSTEM IMAGES

     .-------------------------------------------------------------------.
     |  WARNING: THE SPECIFIED TARGET SYSTEM IMAGE ALREADY EXISTS.        |
     |           THIS REQUEST WILL MERGE ALL THE COMPONENTS FROM THE      |
     |           SOURCE SYSTEM IMAGE INTO THE TARGET SYSTEM IMAGE.        |
     |                                                                    |
     |              * Source System Image - EASTTEST-0001                 |
     |              * Target System Image - EASTPRD1-0001                 |
     |
     |    Enter 'YES' to OVERLAY like-named components .... YES
     |
     '-------------------------------------------------------------------'

     Press Confirm key to confirm request, Cancel key to cancel the request
```

Indicates overlay.

```
SOLVPROD---------- Automation Services : System Image Copy -------EASTTEST-0001
RMDBMS15 SYSTEM IMAGE COPY IN PROGRESS
. System Image  ---------------------------------------------------------------.
|                                                                              |
| From System Name ..... EASTTEST            To System Name ... EASTPRD1       |
|      Version ......... 0001                   Version ....... 0001           |
|                                                                              |
'------------------------------------------------------------------------------'

. Records Processed  ----------------------------------------------------------.
|                                                                              |
| Records Processed .... 76 of 210 36%          Records Written 76             |
| Last Record Class .... 10                     Name........... 22             |
|                                                                              |
'------------------------------------------------------------------------------'

. Percent Complete  -----------------------------------------------------------.
|                                                                              |
| 0                 25                 50               75                100 |
| -------------------------->                                                 |
|                                                                              |
'------------------------------------------------------------------------------'

            SysAvl             Appl                                 O
```

5

Adding Resource Definitions to a System Image

Resource definitions are stored in the knowledge base. You use resource definitions to specify the operations policies required by your organization for the resources.

Refer to your product user's guide for information about any supplied AutoAssist facilities that can help you simplify this task.

This chapter contains the following topics:

- Why Do You Need to Define Resources?
- Implementing Resource Definitions
- Accessing Resource Definitions
- Actions That You Can Perform on a Resource Definition
- Defining a Resource to a System Image
- Handling Timeouts
- Specifying Messages in a Resource Definition
- Extending the Definition of Resource Message Rules
- Executing an NCL Procedure
- Creating User-defined Resource Subclasses
- Using Logical Resources
- Defining Resource Relationships
- Implementing Resource Definitions for System Load Balancing
- Maintaining Resource Definitions

Why Do You Need to Define Resources?

To enable the region to manage the desired state of resources as part of the overall systems operations, you need to define those resources to the knowledge base by using resource definitions.

The resource definitions belong to system images. To work with the definitions, you should have defined at least one image. See Chapter 4, *Working With System Image Definitions*, about how to define images.

Implementing Resource Definitions

Automation Services supports the following classes of resources in addition to those provided by the individual products. The name of the class is in parentheses.

- Internal components (INTNL)
- User-defined resource subclasses (USRCLS)

Additional classes are provided by the products you are using.

The INTNL class resource definitions are reserved for internal use. This chapter describes how to define resources in general.

Using Templates

You define a resource by completing a set of data-entry panels. A set of sample templates for common resources is supplied with your product. You can either use these templates or define your own templates to simplify the task of creating resource definitions.

Using the USRCLS Class

The USRCLS class enables you to specify the operations methods for non-supported resources. A region can then use the methods to manage those resources.

The USRCLS class provides a LOGICAL subclass, which indicates a subclass for logical resources. You can define additional subclasses (see the section, *Creating User-defined Resource Subclasses*, on page 5-43).

Accessing Resource Definitions

You can access resource definitions through one of the following panels:

- Resource Definition panel
- System Image List panel

Accessing Resource Definitions Through the Resource Definition Panel

Enter the **/RADMIN.R.*resource-class-name*** path to access the resource list. For example, to list started tasks, enter **/RADMIN.R.STC**.

The List panel identifies the system image to which the resources belong. If you want to list the resources belonging to another system image, update the System Name and the Version fields.

Accessing Resource Definitions Through the System Image List Panel

Enter the **/RADMIN.I** path to access the System Image List panel. Enter **R** beside the appropriate system image to list the resource definitions in that image.

Actions That You Can Perform on a Resource Definition

You can perform some or all of the following actions from the resource list window:

F4 (Add)	Displays the General Description panel for you to define a new resource.
S, /, or B (Browse)	Displays the Panel Display List window that lists the resource definition panels. When you select a panel, the panel displays the information in the BROWSE (read-only) mode. If you are authorized to update resource definitions, you can press F4 (Edit) from a definition panel to switch to the UPDATE mode.
U (Update)	Displays the Panel Display List window that lists the resource definition panels. When you select a panel, the panel displays the information in the UPDATE mode. You cannot change the system name, system version, or resource name.
C (Copy)	Displays the General Description panel in the COPY mode. You must change at least the system image name, system image version, or resource name. If you copy the definition to another image, any availability map and processes specified in the definition are also copied if they do not exist already in that image. (Global processes are already visible to the image and are not copied to that image.) If you copy the definition to another image and a definition of the same name already exists, you are prompted to either replace the existing definition or cancel the operation. Replacing the definition does not remove any existing parent-child resource relationships.
D (Delete)	Displays a deletion confirmation message. Press ENTER to delete the resource definition. Press F12 (Cancel) to retain the resource definition.
R (Related)	Displays the list of immediate parents and children of the resource definition and enables you to update those relationships.

Defining a Resource to a System Image

All resources are defined in the same way, although the fields on some panels differ according to the class of resource being defined.

You can use variables as data in a resource definition. See Appendix D, *Variables*, for the list of variables that you can use in a definition field.

To define a resource to the system image, press F4 (Add) from the resource list window. A General Description panel is displayed (see Figure 5-2 on page 5-7 for an example). This is the first in a series of panels you use to define the resource to the system image. The series of panels can be different for different classes of resources. The following series is an example:

- General Description—you must complete this panel
- Availability Map
- Activation Details
- Inactivation Details
- Force Inactivation Details
- Display and Heartbeat Details
- Status Monitor Message Details
- State Change Exits
- Automation Log Details
- Owner Details
- Extended Function Exit

Note

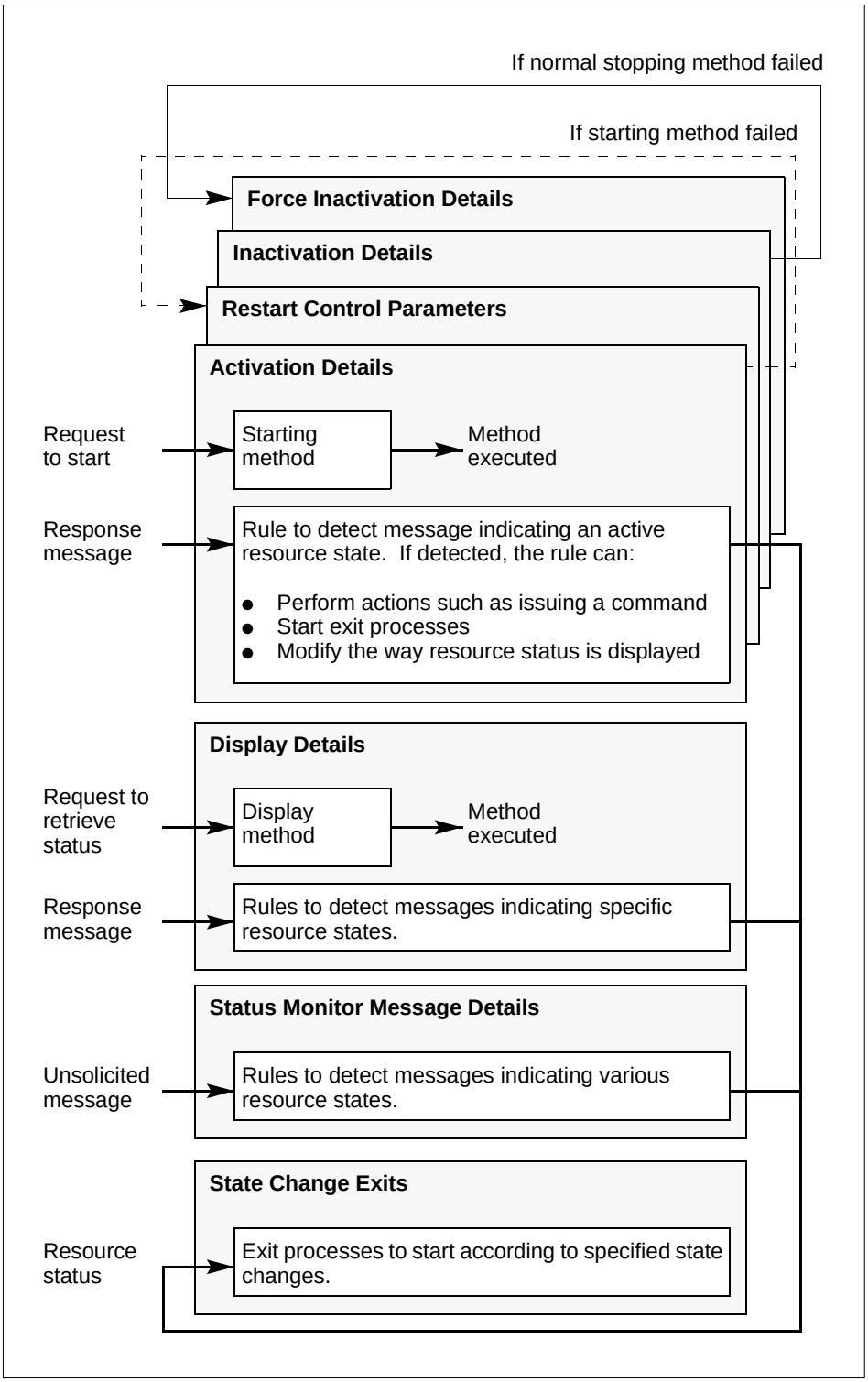
You can facilitate the definition of local resources to a system image by using AutoAssist facilities (if supported by your product).

Note

In the definitions, when you issue a system command, you issue the command by using SYSCMD. You can enter values in system command fields according to the SYSCMD syntax. See the online help or the *Command Reference* for information about the SYSCMD command.

Figure 5-1 shows the main features in a resource definition that enable you to implement the operations policy for a resource.

Figure 5-1. Resource Definition—Main Features



The following example demonstrates the way a resource is added. The example assumes that you are using the SOLVE:Operations Automation product. In the example, you want to add a started task called CICSD to the EASTTEST 0001 system image. Because this is a standard CICS resource, you use a predefined template to complete some of the fields. The template contains such things as the standard commands used to start, stop, and retrieve the status of the resource, and the associated response messages.

Describing the Resource

Use the General Description panel to specify the resource name and type, the name of the VTAM access method control block (ACB) associated with this resource, the operation mode, and a description of the purpose of the resource.

Figure 5-2 shows the completed General Description panel. Because this resource is to be tested, the operation mode is MANUAL, enabling manual control of the resource.

Figure 5-2. Example—General Description Panel

```
SOLVPROD------- ResourceView : Started Task General Description --EASTTEST-0001
Command ===>                                                     Function=ADD

System Name ..+ EASTTEST   Database Version ..+ 0001
Started Task Name ............................ CICSD___
Started Task Type ...........................+ CICS________


ACB Name ............. ________

Operation Mode ......+ MANUAL_____
Short Description .... CICS resource_________________
Long Description ..... ______________________________________________________
                       ______________________________________________________
                       ______________________________________________________
                       ______________________________________________________

+ Template Selection -------------------------------------------------------+
|                                M=Merge O=Override R=Reset S/B=Browse L=List |
|      TemplateName         Copy AvailMap   Copy Process                     |
| ___  CICSESA4____         YES____         YES____                          |
+----------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=File      F4=Save      F5=Related
              F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

You can specify how you want to handle any availability map or processes specified in the template in the Copy AvailMap and Copy Process fields. Valid values are as follows:

No	Do *not* copy.
Replace	Copy; replace if the map or process exists in the system image.
Yes	Copy only if the map or process does *not* exist in the system image.

Note

Global processes are already visible to the resource definition and are not copied into the system image irrespective of the value in the Copy Process field.

In the Template Selection window, you can perform a number of actions by entering the action code beside the TemplateName field. These actions are as follows:

B (Browse)	Displays the contents of a template definition.
L (List)	Displays the list of available templates.
M (Merge)	Merges the values in a template with the existing values in a resource definition (for example, to update a resource definition). Merging does not overwrite existing values, so ensure that you delete the values in the resource definition that are to be replaced. Merging also does not set data that conflicts with existing data (for example, a process name is not set when a command already exists).
O (Override)	Populates the fields in a resource definition with the values in a template (for example, to add a resource definition by using a template as the model).
R (Reset)	Clears the fields in a resource definition, then overrides it with the values in a template.

The M, O, and R actions do not affect the system image, resource name, and description fields specified in a resource definition.

Before you apply the M, O, or R action, you should ensure that the resource name is specified.

Using Resource Templates

A set of sample templates for common resources is supplied (for example, CICS, DB2, and IMS). You can either use these templates or define your own templates to simplify the task of creating resource definitions.

In the example (Figure 5-2), you use the CICSESA4 template to create a CICSD started task definition. You apply the template from the General Description panel. Proceed as follows:

Step 1. On the General Description panel, type CICSD in the Started Task Name field and enter **L** in the Template Selection window.

Step 2. On the Template List panel, enter **O** beside CICSESA4 to apply the template.

Note

If a template is changed, you can use the AP action code to reapply the template to update resource definitions that are using that template. See the section, *Applying Updated Templates*, on page 9-6 for information about how to reapply templates.

Controlling the VTAM ACB of a Resource

Some resources have ACBs. The ACBs must be available before these resources can start successfully. The ACB Name field enables you to control the ACB from the resource definition. If you identify the ACB in the field, the following actions occur automatically during resource startup and shutdown:

- The ACB is activated by the following command before the resource is started:

  `VARY NET,ACT,ID=`*`acb-name`*

- The ACB is inactivated by the following command if the resource is stopped successfully:

  `VARY NET,INACT,ID=`*`acb-name`*`,IMMED`

Specifying the Operation Mode

Specify an operation mode of AUTOMATED, OFF, MANUAL, or IGNORED. During operation, the mode specified here can be restricted by the global operation mode.

The global operation mode can be either AUTOMATED or MANUAL, with the former having a higher rank. If the global operation mode is MANUAL, the resource operation mode of AUTOMATED is forced to MANUAL during operation. See Chapter 14, *Controlling the System Image*, about how to set the global operation mode.

Defining the Availability of the Resource

From the General Description panel, press F8 (Forward) to scroll forward to the Availability Map panel (see Figure 5-3 on page 5-11 for the example). Use this panel to define the changes to the normal availability of the resource. You can attach more than one resource to an availability map.

Leave the Map Name field blank if you want to use the default desired state, which can be either ACTIVE or INACTIVE, depending on a value set in the AUTOIDS parameter group during region initialization.

Note

If you intend to form a parent-child relationship between resource definitions where the availability is determined by the parent, attach the map to the parent definition and define the desired state of the children to be always ACTIVE.

Note

The availability of a service overrides the availability of the resources that make up the service. If the resource always operates as part of a service, let the service handle the availability of the resource. Define the desired state of the resource to be always INACTIVE. When the service starts, it places an ACTIVE desired state override on the resource.

To attach CICSD to an existing map called CICSSCHED, enter **CICSSCHED** in the Map Name field. The map, which might already contain information that affects the availability of the resource, is displayed. Press F10 (Edit Map) to update the timer details. In the example shown in Figure 5-3, all resources attached to this schedule are inactivated at 1700 hours on 23 May 1997 for system maintenance, and reactivated at 0800 hours the next day.

Figure 5-3. Example—Availability Map Panel

```
SOLVPROD---------- ResourceView : STC CICSD Availability Map -----EASTTEST-0001
Command ===>                                                     Function=UPDATE

+- Map Details -----------------------------------------------------------------+
| Map Name .....+ CICSSCHED___                          Expire Delete ... NO     |
| Description ... Availability Map for CICS                                      |
| Timer Execution Control System .........+ N/A      (Service/Shared Images)   |
| Attached Resources ...+ _ CICSA,CICSB                                          |
+--------------------------------------------------------------------------------+
+- Timer Details ----------------------------------------------------------------+
|     Day Date        Time     SVC/Resource Name  Des.State Mode       Status |
|     FRI 23-MAY-1997 17.00.00                    INACTIVE             ON     |
|     SAT 24-MAY-1997 08.00.00                    ACTIVE               ON     |
|                                                                                |
|                                                                                |
|                                                                                |
|                                                                                |
|                                                                                |
|                                                                                |
|                                                                                |
+--------------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=File      F4=Save      F5=Related
 F7=Backward  F8=Forward   F9=Swap     F10=EditMap  F11=Panels   F12=Cancel
```

Note

You can create a new map from the resource definition. You can either name a new map and define it, or access an existing map, change the name, and update the copy. The map is created in the knowledge base when you save the resource definition.

An availability map also enables you to schedule changes to the operation mode and the starting of processes. Chapter 7, *Defining and Maintaining Availability Maps*, describes how to define availability maps.

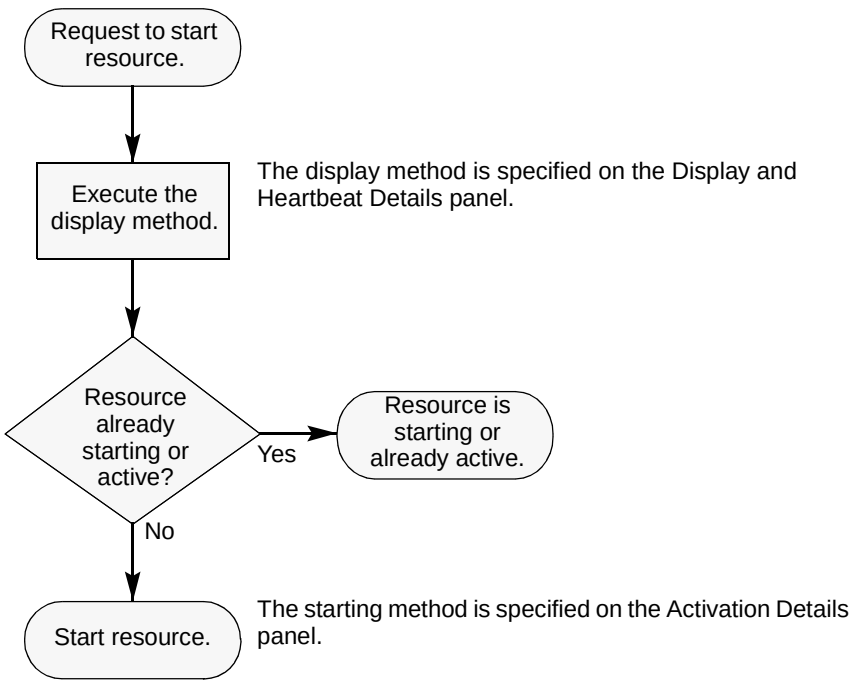
Defining the Activation Details

The activation details specify the following information:

- How to activate the resource
- What to do if the activation is successful
- What to do if the activation is not successful

During automation, the region uses the display method to check the status of the resource that is to be started. If the resource is already starting or has already become active, the starting method is not executed. Figure 5-4 shows the sequence of events that occurs during the starting operation.

Figure 5-4. Resource Starting Operation



Note: A similar sequence of events occurs when a request to stop a resource is processed.

If you want to perform pre-activation processing, specify it on the State Change Exits panel.

From the Availability Map panel, press F8 (Forward) to scroll forward to the Activation Details panel that enables you to supply the information for starting the resource (see Figure 5-5 on page 5-14 for the example). If the starting method is simple, specify a system command, timeout and status change information, and the expected completion message. (See the section, *Handling Timeouts*, on page 5-32 for information about how to specify timeout behavior.)

Note

See the section, *Specifying Messages in a Resource Definition*, on page 5-33 for information about how to simplify the selection of relevant messages for the resource definition.

You can extend the message definition, as described in the section, *Extending the Definition of Resource Message Rules*, on page 5-36.

If the starting method requires more than one command or requires the processing of multiple messages, use a process.

Note

You can use the $NCL process to execute an NCL procedure. See the section, *Executing an NCL Procedure*, on page 5-42 for further information.

A process is a set of steps that are executed to handle a complex task. You can create and modify processes. Chapter 8, *Defining Processes to Implement Complex Operations*, describes how to define processes.

If you use a process, your process must set the return code to one of the following values, to notify the region to set the correct actual state of the resource:

Return Code	Actual State
0	ACTIVE—process is successful.
12	Do nothing. For example, the process might have set the state already by using the SETSTATE macro. **Warning:** You should use this method only if the set state is acceptable to the ACTIVE desired state. The acceptable states are ACTIVE, DEGRADED, and STARTING. If the result of the process is unacceptable, set the return code to, for example, 8 to indicate that the process failed to achieve the desired state.
99	Proceed the same way as if a system command has been issued (that is, wait for the expected completion message and possibly perform timeout processing).
Other	Process is unsuccessful. If timeout processing is specified, perform it immediately. Do not wait for the period specified in the Timeout After field. If timeout processing is not specified, do *not* change the actual state. The region can proceed no further with the automated starting of the resource. The *automation status* is set to FAILED, and the logical state is forced to INERROR.

Note

If you provide *no* command or process for starting the resource, the region cannot start the resource when required. Instead of trying to start the resource, the region places the resource in the MANUAL operation mode.

The exception is when the resource has a type of LOGICAL. See the section, *Using Logical Resources*, on page 5-44 for further information.

Figure 5-5 shows the Activation Details panel with the values supplied by the CICSESA4 template.

Figure 5-5. Example—Activation Details Panel

```
SOLVPROD--------- ResourceView : STC CICSD Activation Details ----EASTTEST-0001
Command ===>                                                   Function=ADD

+ Activation Processing -------------------------------------------------------+
| System Command ....+ S CICSD_____________________________________________ |
|                                                                            |
|       S=ListPnls EF=ExtFilter EA=EvntAct EE=EvntExit ED=ExtDisp DOC=EvntDoc |
|      Expected Activation Completion Message           Pty  Ext   Status    |
| ___  DFHSI1517 CICSD CONTROL___________________________ 000  NO    ACTIVE    |
|                                                                            |
| OR Run Process                     S/B=Browse U=Update I=Interactive L=List |
|      ProcessName  Optional Parameters                                      |
| ___  ________     ________________________________________________________ |
|                   ________________________________________________________ |
|                                                                            |
+ Timeout Processing ----------------------------------------------------------+
|                                                                            |
| Timeout After ...... 00.05.00    On Timeout Assume Status of ..+ __________ |
+ Restart Control Parameters  --------------------------------------------+
|                                                   S/U=Update D=Delete |
| ___ Retry Limit      Time Limit           Command/Process                |
'----------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=File      F4=Save      F5=Related
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

The A (activate) command uses the information on this panel. See the section, *Starting Manual Services and Resources*, on page 13-23 for information about how to use this command.

Defining the Restart Control

Conditions might arise when an automated resource becomes active momentarily only, causing the region to restart the resource. If the condition persists, the region will continue to restart the resource every time the resource deviates from the ACTIVE state. To prevent this situation from continuing unchecked, use the restart control parameters to limit the number of times activation is retried.

The restart control parameters specify the maximum number of restarts that are allowed within a specified time period. The time period starts the first time the resource is activated.

If this retry limit is exceeded, the actual state of the resource is set to FAILED and the operation mode of the resource is set to MANUAL. No further activation is performed.

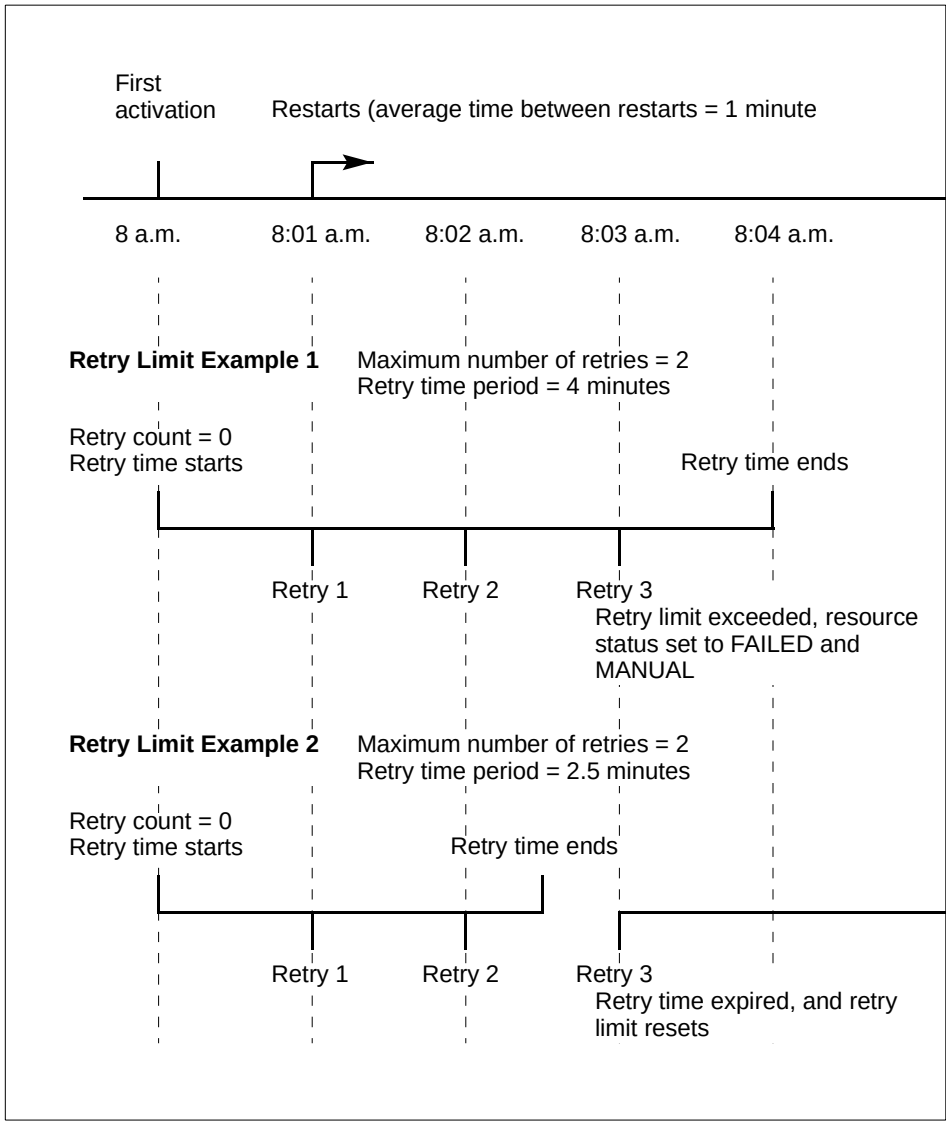
If the retry limit is not exceeded, the next activation that occurs outside the retry time period will reset the retry count and period.

Figure 5-6 shows how different retry limits affect resource restarts. In the first example, the retry limit is exceeded and the resource status is set to FAILED and MANUAL. In the second example, the retry limit is never exceeded and the restarts will continue unchecked.

Caution

Ensure that you allow a long enough time period for the retry limit to be exceeded.

Figure 5-6. Effect of Activation Retry Limits on Resource Restarts



If no restarts are required, the retry time period will expire and the parameters will have no effect on resource operations.

Specifying the Restart Control Parameters

To specify the restart control parameters, enter **U** in the Restart Control Parameters window on the Activation Details panel.

You can specify:

- The retry limit
- What action to take if the retry limit is exceeded

After you complete the panel, press F3 (OK) to return to the Activation Details panel.

Figure 5-7 shows the panel with the values supplied by the CICSESA4 template.

Figure 5-7. Example—Restart Control Parameters Panel

```
SOLVPROD--- ResourceView : STC CICSD Restart Control Parameters --EASTTEST-0001
Command ===>                                                   Function=UPDATE

. Restart Control Parameters ------------------------------------------------.
|                                                                            |
| Retry Attempt Limit ... ___      (How many times will Restart be attempted) |
| Retry Time Limit ...... ________ (Time limit for Restart Retries)           |
'----------------------------------------------------------------------------'
. Retry Limit(s) Exceeded ---------------------------------------------------.
|                                                                            |
| Assume Status of ...... FAILED                                             |
|                                                                            |
+ Issue a Command -----------------------------------------------------------+
|                                                                            |
| System Command... ________________________________________________________ |
|                                                                            |
+ OR Run a Process ----------------------------------------------------------+
|                                     S/B=Browse U=Update I=Interactive L=List |
|      ProcessName  Optional Parameters                                      |
| ___  ________     ________________________________________________________ |
|                   ________________________________________________________ |
+----------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=OK
                           F9=Swap                                F12=Cancel
```

Suppressing Restarts

You can prevent the region from restarting a resource. Enter **0** in the Retry Attempt Limit field and a time period in the Retry Time Limit field.

If an attempt to restart the resource occurs within the specified time period, the activation is not performed and the resource state is set to FAILED. The operation mode of the resource is also set to MANUAL.

Effect of Restart Control on Manual Activation

Restart control applies irrespective of how the resource is activated, whether automatically or manually. The first activation resets the retry limit, and subsequent activations contribute to the retry limit.

Defining the Inactivation Details

The inactivation details specify the following information:

- How to inactivate the resource normally
- What to do if the inactivation is successful
- What to do if the inactivation is not successful

During automation, the region uses the display method to check whether the resource that is to be stopped is already inactive. If the resource is already inactive, the stopping method is not executed.

From the Activation Details panel, press F8 (Forward) to scroll forward to the Inactivation Details panel that enables you to supply the information for stopping the resource (see Figure 5-8 on page 5-19 for the example). If the stopping method is simple, specify a system command, timeout and status change information, and the expected completion message. (See the section, *Handling Timeouts*, on page 5-32 for information about how to specify timeout behavior).

Note

See the section, *Specifying Messages in a Resource Definition*, on page 5-33 for information about how to simplify the selection of relevant messages for the resource definition.

You can extend the message definition, as described in the section, *Extending the Definition of Resource Message Rules*, on page 5-36.

If the stopping method is complex, use a process.

Note

You can use the $NCL process to execute an NCL procedure. See the section, *Executing an NCL Procedure*, on page 5-42 for further information.

If you use a process, your process must set the return code to one of the following values, to notify the region to set the correct actual state of the resource:

Return Code	Actual State
0	INACTIVE—process is successful.
12	Do nothing. For example, the process might have set the state already by using the SETSTATE macro. **Warning:** You should use this method only if the set state is acceptable to the INACTIVE desired state. The acceptable states are INACTIVE and STOPPING. If the result of the process is unacceptable, set the return code to, for example, 8 to indicate that the process failed to achieve the desired state.
99	Proceed the same way as if a system command has been issued (that is, wait for the expected completion message and possibly perform timeout processing).
Other	Process is unsuccessful. If timeout processing is specified, perform it immediately. Do not wait for the period specified in the Timeout After field. If timeout processing is not specified, do *not* change the actual state. The region can proceed no further with the automated stopping of the resource. The *automation status* is set to FAILED, and the logical state is forced to INERROR.

Note

If you provide *no* command or process for stopping the resource, the region cannot stop the resource when required. Instead of trying to stop the resource, the region places the resource in the MANUAL operation mode.

The exception is when the resource has a type of LOGICAL. See the section, *Using Logical Resources*, on page 5-44 for further information.

Figure 5-8 shows the Inactivation Details panel with the values supplied by the CICSESA4 template. The template specifies a forced stop if the CICSD started task does not stop within five minutes of the stop command being issued.

Figure 5-8. Example—Inactivation Details Panel

```
SOLVPROD-------- ResourceView : STC CICSD Inactivation Details ---EASTTEST-0001
Command ===>                                                     Function=ADD

. Inactivation Processing ----------------------------------------------------.
|                                                                              |
| System Command ....+ F CICSD,CEMT PER SHUT_________________________________ |
|                                                                              |
|       S=ListPnls EF=ExtFilter EA=EvntAct EE=EvntExit ED=ExtDisp DOC=EvntDoc |
|      Expected Inactivation Completion Message         Pty  Ext   Status     |
| ___  DFHKE1799 CICSD TERMINATION_______________________ 000  NO    INACTIVE   |
|                                                                              |
| OR Run Process                                                               |
|                                     S/B=Browse U=Update I=Interactive L=List |
|      ProcessName  Optional Parameters                                        |
| ___  ________     ___________________________________________________________ |
|                   ___________________________________________________________ |
|                                                                              |
+ Timeout Processing ---------------------------------------------------------+
|                                                                              |
| Timeout After ..... 00.05.00   On Timeout Assume Status of ....+ __________ |
|                                OR Try Force Inactivation ....... YES         |
+------------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=File      F4=Save      F5=Related
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

The T (terminate) command uses the information on this panel. See the section, *Stopping Manual Services and Resources Normally,* on page 13-23 for information about how to use this command.

Defining the Forced Inactivation Details

The forced inactivation details specify the following information:

- How to force inactivate the resource
- What to do if the forced inactivation is successful
- What to do if the forced inactivation is not successful

During automation, the region uses the display method to check whether the resource that is to be stopped is already inactive. If the resource is already inactive, the stopping method is not executed.

From the Inactivation Details panel, press F8 (Forward) to scroll forward to the Force Inactivation Details panel (see Figure 5-9 on page 5-21 for the example). If the forced method is simple, specify a system command, timeout and status change information, and the expected completion message. (See the section, *Handling Timeouts*, on page 5-32 for information about how to specify timeout behavior).

Note

See the section, *Specifying Messages in a Resource Definition*, on page 5-33 for information about how to simplify the selection of relevant messages for the resource definition.

You can extend the message definition, as described in the section, *Extending the Definition of Resource Message Rules*, on page 5-36.

If the method is complex, use a process.

Note

You can use the $NCL process to execute an NCL procedure. See the section, *Executing an NCL Procedure*, on page 5-42 for further information.

If you use a process, your process must set the return code to one of the following values, to notify the region to set the correct actual state of the resource:

Return Code	Actual State
0	INACTIVE—process is successful.
12	Do nothing. For example, the process might have set the state already by using the SETSTATE macro. **Warning:** You should use this method only if the set state is acceptable to the INACTIVE desired state. The acceptable states are INACTIVE and STOPPING. If the result of the process is unacceptable, set the return code to, for example, 8 to indicate that the process failed to achieve the desired state.
99	Proceed the same way as if a system command has been issued (that is, wait for the expected completion message and possibly perform timeout processing).
Other	Process is unsuccessful. If timeout processing is specified, perform it immediately. Do not wait for the period specified in the Timeout After field. If timeout processing is not specified, do *not* change the actual state. The region can proceed no further with the automated forced stopping of the resource. The *automation status* is set to FAILED, and the logical state is forced to INERROR.

Note

If you provide *no* command or process to force the stopping of the resource, the region cannot force the resource to stop when required. Instead of trying to force the resource to stop, the region places the resource in the MANUAL operation mode.

The exception is when the resource a type of LOGICAL. See the section, *Using Logical Resources*, on page 5-44 for further information.

Figure 5-9 shows the Force Inactivation Details panel with the values supplied by the CICSESA4 template.

Figure 5-9. Example—Force Inactivation Details Panel

```
SOLVPROD----- ResourceView : STC CICSD Force Inactivation Details EASTTEST-0001
Command ===>                                                     Function=ADD

. Force Inactivation Processing ---------------------------------------------.
|                                                                            |
| System Command ....+ C CICSD_______________________________________________ |
|                                                                            |
|       S=ListPnls EF=ExtFilter EA=EvntAct EE=EvntExit ED=ExtDisp DOC=EvntDoc |
|      Expected Force Inactivation Completion Message   Pty  Ext   Status    |
| ___  IEF404I CICSD - ENDED_____________________________ 000  NO    INACTIVE  |
|                                                                            |
| OR Run Process                                                             |
|                                   S/B=Browse U=Update I=Interactive L=List |
|      ProcessName  Optional Parameters                                      |
| ___  ________     ________________________________________________________ |
|                   ________________________________________________________ |
|                                                                            |
+ Timeout Processing ---------------------------------------------------------+
|                                                                            |
| Timeout After ..... 00.03.00   On Timeout Assume Status of ....+ __________ |
|                                                                            |
'----------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=File      F4=Save      F5=Related
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

The TF (terminate by force) command uses the information on this panel. The T (terminate) command might also use this information if a timeout action of forced stop is specified on the Inactivation Details panel. See the section, *Stopping Manual Services and Resources Normally*, on page 13-23 for information about how to use these commands.

Defining the Display and Heartbeat Details

The display and heartbeat details specify how to find out about the status of the resource.

From the Force Inactivation Details panel, press F8 (Forward) to scroll forward to the Display and Heartbeat Details panel (see Figure 5-10 on page 5-23 for the example). This panel enables you to specify the display method for status checking. If the display method is simple, specify a system command, heartbeat interval, and the expected response message.

Note

See the section, *Specifying Messages in a Resource Definition*, on page 5-33 for information about how to simplify the selection of relevant messages for the resource definition.

You can extend the message definitions, as described in the section, *Extending the Definition of Resource Message Rules*, on page 5-36.

If the display method is complex, use a process.

Note

You can use the $NCL process to execute an NCL procedure. See the section, *Executing an NCL Procedure*, on page 5-42 for further information.

You can specify a system command or a process without specifying the heartbeat interval.

If you use a process, your process must set the return code to one of the following values, to notify the region to set the correct actual state of the resource:

Return Code	Actual State
0	ACTIVE.
1	STARTING.
2	STOPPING.
3	DEGRADED.
5	INACTIVE.
6	FAILED.
12	Do nothing. For example, the process might have set the state already by using the SETSTATE macro.
Other	UNKNOWN.

Note

Before the region starts or stops a resource, the region executes the display method to check the actual state of the resource.

The heartbeat interval enables you to set up regular checking of the resource status.

Caution

The heartbeat feature increases CPU consumption. Use this feature only for resources that are liable to change state without an accompanying message.

Figure 5-10 shows the Display and Heartbeat Details panel with the values supplied by the CICSESA4 template.

Figure 5-10. Example—Display and Heartbeat Details Panel

```
SOLVPROD--- ResourceView : STC CICSD Display and Heartbeat Details EASTTEST-000
Command ===>                                                     Function=ADD

+ Resource Display Processing (Command) -------------------------------------+
| System Command ....+ D J,CICSD_____________________________________________ |
| Heartbeat Interval . ________                                               |
| Status Indication Message                                                   |
|      S=ListPnls EF=EvntFilter EA=EvntAct EE=EvntExit ED=ExtDisp DOC=EvntDoc |
|      Message Text                                          Status      Ext  |
| ___  CICSD    CICSD_____________________________________ ACTIVE      NO   |
| ___  CICSD NOT FOUND____________________________________ INACTIVE    NO   |
| ___  ___________________________________________________ STARTING          |
| ___  ___________________________________________________ STOPPING          |
| ___  ___________________________________________________ DEGRADED          |
| ___  ___________________________________________________ FAILED            |
| ___  ___________________________________________________ UNKNOWN           |
+-----------------------------------------------------------------------------+
+ OR Run Display Process -----------------------------------------------------+
|                                 S/B=Browse U=Update I=Interactive L=List |
|      ProcessName  Optional Parameters                                       |
| ___  ________     ________________________________________________________ |
+-----------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=File      F4=Save      F5=Related
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Generally, the supplied CHECKALL, CHK (check), and D (display) commands use the method on this panel when checking or displaying the resource status. See Appendix A, *Commands*, for information about these commands. The exceptions are resources that have their own special D commands. You can identify these exceptions from the list of command definitions. To access the list, enter the **/ASADMIN.C** path.

Defining the Status Monitor Message Details

The status monitor message details specify rules that can be triggered by messages to perform actions.

From the Display and Heartbeat Details panel, press F8 (Forward) to scroll forward to the Status Monitor Message Details panel (see Figure 5-11 on page 5-25 for the example). This panel displays two windows: the Activate Inactivate Messages window and the Status Monitor Messages window.

The Activate Inactivate Messages window displays the messages previously specified on the Activation Details, Inactivation Details, and Force Inactivation Details panels.

The Status Monitor Messages window enables you to set up message rules to perform actions that are message dependent. The information defined on this panel includes the message identifier and text, the message rule priority, and the effect of the rule on the resource state. You can define up to 97 lines of message rules.

Note

You can use special message prefixes for special types of messages. See the section, *Using Special Message Prefixes*, on page 5-35.

See the section, *Specifying Messages in a Resource Definition*, on page 5-33 for information about how to simplify the selection of relevant messages for the resource definition.

The rule priority is only significant when rules overlap. Priorities are organized in descending numeric order. For example, the value 10 indicates a higher priority than the value 20. If rules overlap, only the rule with the highest priority is processed. If overlapping rules have the same priority value, the most specific rule has the highest priority. For a set of overlapping rules, the rules are sorted in the order of decreasing priority. You can re-sort the rules after you add or update the rules.

The modifiable Status fields are optional. If a Status field is blank, the message rule does not trigger a state change.

The region uses the rules to monitor the messages for the particular resource. If a rule is triggered, the resource state is updated according to the value in the Status column.

You can extend the message definitions, as described in the section, *Extending the Definition of Resource Message Rules*, on page 5-36.

Figure 5-11 shows the Status Monitor Message Details panel with the values supplied by the CICSESA4 template.

Figure 5-11. Example—Status Monitor Message Details Panel

```
SOLVPROD--- ResourceView : STC CICSD Status Monitor Message Details EASTTEST-00
Command ===>                                                   Function=ADD

+-------------------------------------------------------------------------------+
|      Activate Inactivate Messages                      Pty  Ext   Status     |
| ___  DFHSI1517 CICSD CONTROL___________________________ 000  NO    ACTIVE     |
| ___  DFHKE1799 CICSD TERMINATION_______________________ 000  NO    INACTIVE   |
| ___  IEF404I CICSD - ENDED_____________________________ 000  NO    INACTIVE   |
'-------------------------------------------------------------------------------'
.-------------------------------------------------------------------------------.
|  S=LstPnls EF=ExtFilter EA=EvntAct EE=EvntExit ED=ExtDisp DOC=EvntDoc D=Del |
|      Status Monitor Messages                           Pty  Ext   Status     |
| ___  $HASP100 CICSD    ON______________________________ ___  NO    STARTING__ |
| ___  $HASP373 CICSD    STARTING________________________ ___  NO    STARTING__ |
| ___  $HASP395 CICSD    ENDED___________________________ ___  NO    INACTIVE__ |
| ___  $HASP396 CICSD    TERMINATED______________________ ___  NO    FAILED____ |
| ___  DFHAP0001_________________________________________ ___  YES   DEGRADED__ |
| ___  DFHAP0002_________________________________________ ___  YES   DEGRADED__ |
| ___  DFHAP0003_________________________________________ ___  YES   DEGRADED__ |
| ___  DFHAP0004_________________________________________ ___  YES   DEGRADED__ |
| ___  DFHAP0005_________________________________________ ___  YES   DEGRADED__ |
+-------------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=File      F4=Save      F5=Related   F6=SortMsgs
 F7=Backward  F8=Forward   F9=Swap     F10=ScrlLst  F11=Panels   F12=Cancel
```

Use this panel to handle unplanned events such as degraded performance and failures. Define the recovery procedures for the unplanned events by using the extended message rule definition feature.

Extended Message Definition Considerations

Normally, actions associated with the status monitor message rules are performed only if the resource is in the AUTOMATED operation mode. This condition is set in the AUTOIDS region parameter group. To access the list of parameter groups, enter the **/ICS** shortcut.

The information specified on the Define Extended Display Attribute panel, however, is always acted on irrespective of the operation mode.

Using the RECOVERED Actual State

The RECOVERED actual state enables the recognition of messages that are the results of recovery from a DEGRADED or FAILED state.

DEGRADED and FAILED are special actual states consisting of a base state and an applied flag as follows:

Actual State	Flag	Base Actual State When the Flag Is Applied
DEGRADED	DEGRADED	ACTIVE
FAILED	FAILED	Previous actual state

Note

The base actual state can be changed by the automation engine. See the next section, *Desired State Management Considerations*, on page 5-27 for information about how the automation engine reacts to the desired state of a resource.

To detect a recovery condition, you should use the RECOVERED actual state. When the state is triggered, it removes the DEGRADED or FAILED flag, and sets the base state to ACTIVE.

Example

You want to set the actual state of a tape unit to DEGRADED when a tape is to be mounted on the unit. On completion of the operation, you want to reset the state of the unit.

Figure 5-12 shows an example. The IEF233A rule indicates that a tape is to be mounted. The $DN-IEF233A rule indicates that a delete operator message (DOM) notification is received for the tape mounting request.

Figure 5-12. Using the RECOVERED Actual State

```
SOLVPROD----- ResourceView : TAPE 0880 Status Monitor Message Details ---------
Command ===>                                                   Function=BROWSE

.-----------------------------------------------------------------------------.
|      Activate Inactivate Messages                     Pty  Ext   Status     |
| ___  IEE302I 0880 ONLINE                              000  NO    ACTIVE     |
| ___  IEF281I 0880 NOW OFFLINE                         000  NO    INACTIVE   |
| ___  IEF281I 0880 NOW OFFLINE - DEVICE IS BOXED       000  NO    INACTIVE   |
'-----------------------------------------------------------------------------'
.-----------------------------------------------------------------------------.
|      S=ListPnls EF=ExtFilter EA=EvntAct EE=EvntExit ED=ExtDisp DOC=EvntDoc  |
|      Status Monitor Messages                          Pty  Ext   Status     |
| ___  IEF233A M 0880                                        YES   DEGRADED   |
| ___  $DN-IEF233A M 0880                                    YES   RECOVERED  |
```

Desired State Management Considerations

The automation engine acts as follows when the actual state of a resource changes to an actual state that is not the same as its desired state:

If the actual state changes to …	**The automation engine …**
ACTIVE or INACTIVE	Attempts to bring the resource to the desired state.
DEGRADED	Attempts to bring the base state of the resource to match that of the desired state.
FAILED	Does not attempt recovery unless recovery actions are specified in the message rule.
STARTING or STOPPING	Takes no action unless actions are specified in the message rule.
UNKNOWN	Invokes the specified display processing to attempt to determine the state.

Implementing the State Change Exits

From the Status Monitor Message Details panel, press F8 (Forward) to scroll forward to the State Change Exits panel. This panel enables you to specify two types of exit processes:

- A process that executes before the starting method specified on the Activation Details panel is performed.
- Processes that execute when specified state changes occur. For example, if a resource fails, you may want to invoke a process that writes a problem report. You can specify a process to execute when the actual state, the desired state, or the logical state of the resource changes.

Figure 5-13 shows an example in which the State Change Exits panel specifies that a process be executed when the CICSD started task fails.

Figure 5-13. Example—State Change Exits Panel

```
SOLVPROD-------- ResourceView : STC CICSD State Change Exits -----EASTTEST-0001
Command ===>                                                     Function=ADD

. State Change Exits -------------------------------------------------------.
|                      S/B=Browse U=Update I=Interactive L=List P=Parameters |
|     Invoked                     Process  Parameters                       |
| ___ BEFORE ACTIVATION           ________                                  |
|     State   Change    Change                                              |
|     Type    From      To        Process  Parameters                       |
| ___ ACTUAL_ ANY______ FAILED___ PROBSOLV ACB=PROB                         |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
| ___ _______ _________ _________ ________                                  |
'---------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Defining the Logging Details

From the State Change Exits panel, press F8 (Forward) to scroll forward to the Automation Log Details Panel. This panel displays information about the size of the temporary log for the resource (called a *transient log*), the destination of the logged information, and the type of information logged.

Figure 5-14 shows the Automation Log Details panel with the values assigned by the CICSESA4 template.

Figure 5-14. Example—Automation Log Details Panel

```
SOLVPROD------- ResourceView : STC CICSD Automation Log Details --EASTTEST-0001
Command ===>                                                     Function=ADD

Resource Log Controls

  Log Table Size ............. 0150
  Log to Automation Log ...... YES
  Log to Console ............. NO
  Log to OCS Window .......... NO

Resource Log Content Controls

  Log All System Msgs ........ NO
  Log Internal Audit Trail ... NO









 F1=Help      F2=Split     F3=File      F4=Save      F5=Related
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Table 5-1 lists the options, their values, and their meanings.

Table 5-1. Log Options

Options	Available Values	Available Values
Log Table Size	0 through 9999	The total number of messages that the transient log can hold. The default is set in the AUTOTABLES parameter group during the initialization of the region. The transient log is kept in memory. A large size can affect extended memory allocation. The transient log is cyclic. That is, if the log is full, a new message displaces the oldest message.

(Sheet 1 of 2)

Table 5-1. Log Options

Options	Available Values	Available Values
Log to Automation Log	NO or YES	Specifies whether to write transient log messages to the activity log. Initial value: NO
Log to Console	NO or YES	Specifies whether to send messages to the system console as WTO messages. Select this option only for resources that are started before VTAM and if you want to debug the resource definitions. **Note** If non-VTAM terminal support is configured, you can use the local non-SNA terminal to view system activities before VTAM becomes available. Initial value: NO
Log to OCS Window	NO or YES	Specifies whether to send messages to users who are monitoring from a message monitor. Initial value: NO
Log All System Msgs	NO or YES	Specifies whether to log all messages for a resource. If the value is NO, the region logs only messages that match the message rules in the resource definition. Messages are written to the transient log and to any destinations specified in the previous options. Initial value: NO
Log Internal Audit Trail	NO or YES	Specifies whether to log the internal audit trail. The audit trail is a detailed log of the processing and actions performed in the region, and is useful for debugging the resource definition. **Note** The option creates a high volume of message flow. Turn on the option only if required. The audit trail is logged to the transient log and any destinations specified in the previous options. Initial value: NO

(Sheet 2 of 2)

Specifying the Owner Details

From the Automation Log Details panel, press F8 (Forward) to scroll forward to the Owner Details panel. This panel enables you to identify up to two people who can be contacted if the resource has operational problems. The template does not assign this information.

Implementing the Extended Function Exit

From the Owner Details panel, press F8 (Forward) to scroll forward to the Extended Function Exit panel. This panel enables you to provide additional operator functions. Specify the exit NCL procedure that provides these functions. The procedure is invoked when an operator issues the XF command against the resource.

The extended function exit NCL procedure has access to variables with the prefix ZRM. Appendix D, *Variables*, describes these variables.

Handling Timeouts

You can specify how long to wait for the response to an action in the Timeout After field on the following panels:

- Activation Details panel
- Inactivation Details panel
- Force Inactivation Details panel

Normally, the response to the action sets the actual state of the resource to the state specified in the Status column.

However, if for any reason the expected response to the action does not come, then, depending on the timeout settings, the following occurs:

- If you have *not* specified a value in the Timeout After field, the region keeps waiting for a response until a significant event changes the status of the resource.

  The exception is the timeout for a resource of the LOGICAL type. For a logical resource, the default value for the Timeout After field is 0 (that is, no waiting).

- If you specified a value in the Timeout After field and you specified the action to take when timeout occurs, the region takes the specified action. The action sets the resource to a specified actual state. On the Inactivation Details panel, you can, instead of specifying the actual state to set, specify that the resource be stopped by force according to the forced inactivation details.

- If you specified a value in the Timeout After field but you did *not* specify any action, the region performs the display processing specified on the Display and Heartbeat panel.

  Note

  For a logical resource, display processing is not applicable. The actual state of the logical resource is set directly to the expected result of the action. A starting action results in an actual state of ACTIVE; a stopping action results in an actual state of INACTIVE.

  If the response to the action indicates that the desired state is satisfied, the region sets the correct actual state.

  If the response to the action indicates a state other than the desired state, the region can proceed no further with the automated process. The region does *not* change the actual state of the resource, but sets the *automation status* for the resource to FAILED and forces the logical state of the resource to INERROR.

Specifying Messages in a Resource Definition

While defining a resource, you might need to specify messages that indicate different resource states on the following panels:

- Activation Details panel
- Inactivation Details panel
- Force Inactivation Details panel
- Display and Heartbeat Details panel
- Status Monitor Message Details panel

Enter the following querying codes in a message field to help you specify the messages:

- Enter **?** to obtain a list of suggested messages.
- Enter **??** to obtain a list of all the messages encountered on the local system.
- Enter **???** to obtain a list of the messages in the transient log for the resource. This feature simplifies message selection, because a transient log contains only messages that are relevant to the resource.

Using Messages in the Transient Log

When you are working on a resource definition, use the transient log of a resource as a source of resource-specific messages.

Before you use the ??? action, you should ensure the following:

- On the Automation Log Details panel of the resource definition, the value in the Log All System Msgs field is YES.
- The definition has been saved in the knowledge base.
- The system image that contains the resource is active.

To use the ??? action, proceed as follows:

Step 1. Enter the **/RMON** shortcut to access the status monitor. If you are using a filter, you might need to change the filter in order to see the resource. (Use the FILTER command to list the defined filters.)

Step 2. Find the resource, ensure that the operation mode is IGNORED or MANUAL, then issue the A (activate) and the T (terminate) commands manually to start and stop the resource. Issue the D (display) command when the resource becomes active and when the resource becomes inactive. These actions populate the transient log with messages.

Step 3. Enter **DB** beside the resource. The Panel Display List panel for the resource definition is displayed.

Step 4. Select the required panel, and enter **???** in a message field on the panel. The Transient Log Browse panel is displayed.

Step 5. Browse the transient log, and select the appropriate message. The resource definition panel is redisplayed with the selected message in the message field.

Using Special Message Prefixes

Normally, a resource definition in an active system image recognizes only messages for the corresponding resource. However, three special message prefixes are available that enable a resource definition to handle special types of local messages. You *cannot* combine prefixes.

$AA-

A resource definition can generate a user-defined resource event. For another resource definition to recognize the event message, you must prefix the message by $AA- when you specify the message rule in that definition.

For example, `ALERT - VTAM AVAILABLE` is specified as the event message on the Define Event Related Action panel of a resource definition. To enable the resource you are defining to recognize that message, you must specify `$AA-ALERT - VTAM AVAILABLE` as the message text to look for.

$DN-

A message can be a delete operator message (DOM) notification. For a resource definition to recognize a DOM notification, you must prefix the message by $DN-.

For example, specifying `$DN-IEF233A M 380` enables a resource definition to recognize the DOM notification for the tape mounting request message `IEF233A M 380`.

$MN-

A resource definition recognizes messages that relate to the resource itself only. For a resource definition to recognize messages from other resources and from the operating system, you must prefix the messages by $MN-.

For example, you might want to detect the starting and the stopping of a batch job that performs system backup. Detecting those messages enables you to stop and restart a resource, as required, by changing the desired state of the resource through the SVCMD macro in a process.

Extending the Definition of Resource Message Rules

You can extend the definition of any message rule in a resource definition (as entered on the Activation, Inactivation, Force Inactivation, Display and Heartbeat, and Status Monitor Message Details panels).

To extend a rule, enter **S** beside the message. A selection panel is displayed, listing the following extended message definition panels:

- Define Extended Filter panel
- Define Event Related Actions panel
- Define Event Exits panel
- Define Event Display Attribute panel
- Event Documentation panel

Defining an Extended Message Filter

Use the Extended Filter Definition panel to specify the criteria for analyzing message text. A message is acceptable if it meets the criteria you specify here. For example, the same message may indicate different resource states, depending on the event that generates the message.

Note

If you want to capture a message that has leading blanks, you do not need to specify the leading blanks on the message filter panel. However, on the Extended Message Filter panel, absolute position is important so leading blanks must be counted when using start position of text.

In the following example, the region receives $HASP605 messages when an operator enters a display command against an initiator called INIT1. The $HASP605 messages show the initiator identifier, what activity it is performing, the job it is processing, and the job classes that it processes. The text of the $HASP605 messages can be analyzed to determine the actual state of INIT1 based on the activity INIT1 is performing.

Figure 5-15 shows the definition of the $HASP605 messages on the Display and Heartbeat Details panel.

Figure 5-15. Example—Defining Message Filters: 1

```
SOLVPROD----- ResourceView : INIT 1 Display and Heartbeat Details EASTTEST-0001
Command ===>                                                   Function=ADD

+ Resource Display Processing (Command) -------------------------------------+
| System Command ....+ $DI1                                                  |
| Heartbeat Interval . ________                                              |
| Status Indication Message                                                  |
|      S=ListPnls EF=EvntFilter EA=EvntAct EE=EvntExit ED=ExtDisp DOC=EvntDoc |
|      Message Text                                          Status      Ext |
| ___  $HASP605    INIT  1                                   ACTIVE      YES |
| ___  $HASP605    INIT  1                                   INACTIVE    YES |
| ___  ________________________________________________       STARTING        |
| ___  ________________________________________________       STOPPING        |
| ___  ________________________________________________       DEGRADED        |
| ___  ________________________________________________       FAILED          |
| ___  ________________________________________________       UNKNOWN         |
+----------------------------------------------------------------------------+
+ OR Run Display Process -----------------------------------------------------+
|                                     S/B=Browse U=Update I=Interactive L=List |
|      ProcessName  Optional Parameters                                      |
| ___  ________     ___________________________________________________      |
+----------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=Exit      F4=Edit      F5=Related
 F7=Backward  F8=Forward   F9=Swap                   F11=Panels
```

You can define extended filters for each occurrence of the message to distinguish which events determine which status. Enter **EF** beside a message to display the Define Extended Filter Definitions panel. Figure 5-16 shows the extended filter definition for the $HASP605 message that indicates an ACTIVE actual state for INIT1.

Figure 5-16. Example—Defining Message Filters: 2

```
SOLVPROD-- ResourceView : INIT 1 Define Extended Filter Definitions EASTTEST-00
Command ===>                                                   Function=UPDATE

Extended Filter Definition

  Message Text .......... $HASP605    INIT  1                  ____________

  Wildcard Character .... *__

+ Message Text Analysis -----------------------------------------------------+
|                                                                            |
|   Strt Word      Scan                                                      |
|   Pos  Num  Opr  Text                                                      |
| 1 ___  004  EQ   ACTIVE                                                    |
| 2 ___  004  EQ   INACTIVE                                                  |
| 3 ___  ___  ___  ________________________________________________          |
| 4 ___  ___  ___  ________________________________________________          |
| 5 ___  ___  ___  ________________________________________________          |
|                                                                            |
| Expression e.g (1 and (2 or 3)) 1 OR 2                                     |
+----------------------------------------------------------------------------+

 F1=Help      F2=Split     F3=OK
              F8=Forward   F9=Swap                 F11=Panels   F12=Cancel
```

The following message indicates that the initiator is processing a job:

```
$HASP605    INIT  1 ACTIVE   JOB06931 C=A
```

The following message indicates that the initiator is waiting for the arrival of the next job:

```
$HASP605    INIT  1 INACTIVE ******** C=A
```

Both the above messages indicate an ACTIVE actual state because the initiator is either being used or is available for use.

When the region receives a $HASP605 message, it scans the message to see if the text starting at the first position of the fourth word is ACTIVE or INACTIVE. If it is either ACTIVE or INACTIVE, the actual state of INIT1 is ACTIVE.

Appendix F, *Message Text Analysis*, provides further information about message text analysis.

Defining Event-related Actions

Use the Define Event Related Actions panel to define particular actions that the region performs when it receives a particular message. The actions are the following:

- Change the operation mode of the resource
- Issue a system command, reply to a WTOR message, or generate a resource event—use the $AA- message prefix when specifying a message rule in a resource definition if you want the definition to recognize resource events
- Log a user-defined message
- Execute a process

Note

You can use the $NCL process to execute an NCL procedure. See the section, *Executing an NCL Procedure*, on page 5-42 for further information.

Implementing Event Exits

Use the Define Event Exits panel to define state change exits, problem exits, or other exits that are specific to this message rule. These exits are invoked when a message satisfying the rule is received (as opposed to the exits specified on the State Change Exits panel that are invoked by a change in resource state). For example, if the rule is for a message generated by an event that causes the performance of the resource to degrade, you may want to specify an exit that logs the degradation in a problem management application.

Defining Extended Display Attributes

Use the Define Extended Display Attribute panel to do the following:

- Override the status monitor fields.
- Override the display attributes.
- Define keyword that can be used by status monitor filters.
- Define information that can be displayed on icons.

Figure 5-17 is an example of the Define Extended Display Attribute panel.

Figure 5-17. Example—Define Extended Display Attribute Panel

```
SOLVPROD--- ResourceView : INIT 1 Define Extended Display Attribute EASTTEST-00
Command ===>                                                   Function=BROWSE

+ Extended Display (EXTDISP) ------------------------------------------------+
| INACTIVE CLASS &ZMSGWORD6                                                   |
+-----------------------------------------------------------------------------+
+ Extended Display Attributes ------------------------------------------------+
|                                                                             |
|   Intensity    Colour       Highlight  Use on Graphic Monitor?    Severity  |
|   LOW          GREEN        NONE           (Change Icon Colour?)     (1-9)  |
+-----------------------------------------------------------------------------+
. User Defined Filter Keyword -----------------------------------------------.
|                                                                             |
| Keyword Value ...              (Use USERKEYW when defining Filters)         |
'-----------------------------------------------------------------------------'
+ User Defined Variables -----------------------------------------------------+
|                                                                             |
| Var    Value                            Var    Value                        |
|      =                                       =                              |
|      =                                       =                              |
|      =                                       =                              |
+-----------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=Exit                                F7=Backward
 F8=Forward   F9=Swap                  F11=Panels
```

Overriding the Status Monitor Fields

Use the Extended Display field to define the text to be overlaid on the status monitor line for this resource. The override text can pick up specified words from the original message if you specify &ZMSGWORD*n*. For example, if you specify an extended display of ACTIVE CLASS=&ZMSGWORD9, the override text displays the class as the ninth word in the original message.

If you leave this field blank and the rule triggers a state change, the status monitor line resets to normal display. If the rule does not trigger a state change and you want the monitor line reset to normal display, specify ##RESET## in the field.

You can use the Severity field in the Extended Display Attributes window to determine whether an extended display can be overwritten by another extended display. When an extended display is invoked, it can overwrite a display of equal or lesser severity only. The field ensures that the monitor displays the most severe condition. The greater the value in this field, the lesser the severity. For example, 9 is less severe than 1.

An operator, using the status monitor, can enter the primary command EXTDISP ON to display override text on the monitor or EXTDISP OFF to display standard message text. The default is to display override text.

Overriding the Display Attributes

Use the Extended Display Attributes window to specify the display attributes and how an icon on the graphical monitor is affected by the attributes.

- Normally, the display attributes of a status line on a status monitor are determined by the logical state of the relevant resource. By using the fields in this window, you can override the normal display attributes.
- An icon on the graphical monitor displays the status of the resource that is in the worst logical state. The Use on Graphic Monitor? field enables you to indicate whether the specified extended display color should be used on an icon.

  If you do not want the icon to use the specified extended display color, specify **NO** (the default) in the Use on Graphic Monitor? field. For example, if the resource does not require attention, transferring the color to the icon can cause an undesirable change in the icon color.

  If you want the icon to use the specified extended display color, specify **YES** in the Use on Graphic Monitor? field. This setting also forces the state ranking of the resource to equal that of the FAILED logical state. The icon can thus be forced to display this resource even though it is not in the worst logical state. For example, if you rate the condition of a tape mount request very important, specify **YES** in the Use on Graphic Monitor? field.

Using the User-defined Keyword

The keyword is used by status monitor filters to determine whether to display the status of a resource on the basis of the processed message.

For example, you might want to suppress the display of a resource on the status monitor when a particular message is detected. To do this, specify a keyword in the rule for that message and use the keyword as a criterion in a status monitor filter to *not* display the resource. When an operator uses the filter to view the resources on the status monitor, the resource is normally displayed on the monitor. However, if a message arrives and satisfies this message rule, the resource disappears from the monitor because the keyword in the filter tells the monitor *not* to display the resource.

Using User-defined Variables

Use the User Defined Variables window to specify data that can be used within an icon definition. The data is only available when the resource for which the data is defined has the worst status of all resources within an icon.

Documenting the Message Rule

Use the Event Documentation panel to record information about a message rule. For example, you might want to document what causes a particular ABEND message.

Executing an NCL Procedure

Note

You *cannot* use the $NCL process from within another process.

$NCL is a special process definition that is applicable across all defined system images. The $NCL process enables you to execute an existing NCL procedure directly. You do *not* need to define a process for the sole purpose of executing an NCL procedure. The $NCL process return code inherits the return code set by the NCL procedure. The syntax is as follows:

```
$NCL $NCL=ncl-procedure-name    parameter-1=value-1
                                .
                                .
                                .
                                parameter-n=value-n
```

$NCL=*ncl-procedure-name*

Names the NCL procedure. The name is mandatory.

parameter-1=value-1 ... parameter-n=value-n

Supplies parameter values to the NCL procedure. The parameters must be in keyword format (for example, NAME=*resource-name*). The parameter names must not start with $.

Creating User-defined Resource Subclasses

User-defined resource subclasses enable you to categorize resources in a class defined by you. You use resource subclasses to categorize resources that do *not* belong to any of the supplied resource classes.

You can monitor and control the resources in the subclasses from the monitors. You can define resource templates for the resource subclasses.

You use the USRCLS resource class when you define resources for resource subclasses. The User Class field of the resource definition determines the resource subclass. A resource subclass is a resource type in the USRCLS resource class.

Initially, the User Class field has one valid value, LOGICAL (indicating a logical resource). To define resources for your own subclasses, you must first define the subclasses in the User Classes field prompt list. The name of a resource subclass can be up to eight characters long.

Note

When your status monitor displays resources of the USRCLS class, you do *not* see USRCLS as a resource class. The resource subclasses are displayed instead.

Add a new user subclass in the User Classes field prompt list as follows:

Step 1. Enter **?** in the User Class field on the User Class Resource General Description panel. The User Classes field prompt list is displayed.

Step 2. Press F10 (EditList) to update the list.

Step 3. Press F4 (Add). The Field Prompt Entry Definition panel is displayed.

Step 4. Define the new user subclass, then press F3 (File) to add the subclass to the list.

Using Logical Resources

The function of some resources is purely to elicit some sort of activity from another resource (for example, starting or stopping other resources). Other resources might be logical representations of groups of resources. Define these types of resources as the LOGICAL resource type.

Note

The name of a logical resource must contain alphanumeric, @, #, $, ., :, -, (, and) characters only. It must not be a number.

Logical resources respond to the actions of starting, stopping, and forced stopping differently from the other types of resources. Logical resources respond to these actions as follows:

- If no command or process exists for the action, the actual state of the logical resource is set to the expected result of the action. A starting action results in an ACTIVE actual state; a stopping action results in an INACTIVE actual state.
- If a command exists and no timeout processing is specified, the region issues the command and then *immediately sets the actual state of the logical resource to the value of the desired state*.
- If a command exists and timeout processing is specified, the timeout settings determine the final status of the resource. See the section, *Handling Timeouts*, on page 5-32 for information about how to specify timeout behavior.
- If a process exists, the region acts the same way as for other resources (that is, the region responds according to the process return code).

Example

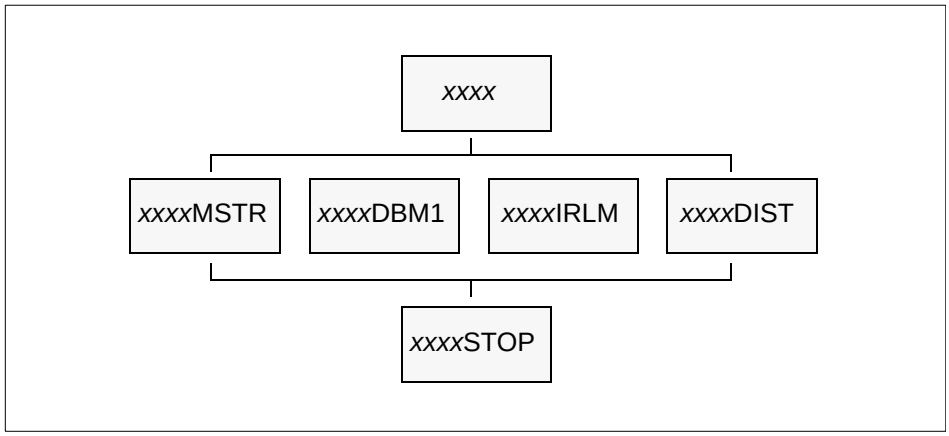
In this example, you set up resource definitions to handle the DB2 subsystem.

Each system can have more than one DB2 subsystem. When issuing a DB2 command, you prefix the command by a subsystem recognition character to differentiate between the different subsystems. Examples of subsystem recognition characters are - and +.

In this example, the DB2 subsystem has a plus sign (+) as the subsystem recognition character.

A DB2 subsystem contains multiple resource components. You can store the DB2 operations methods in a single resource definition. However, if you want the individual components to be visible, you need to create resource definitions for them. Figure 5-18 shows a sample relationship.

Figure 5-18. DB2 Resource Relationships



The resource components *xxxx*MSTR, *xxxx*DBM1, *xxxx*IRLM, and *xxxx*DIST are bounded by the *xxxx* logical resource definition (which contains the starting method) and the *xxxx*STOP logical resource definition (which contains the stopping method).

In normal starting operation, *xxxx* issues the +START DB2 command to start the DB2 subsystem. The DSN9022I message is visible to the *xxxx*MSTR resource definition, which sends an alert indicating that the DB2 subsystem has become active. The alert is visible to *xxxx*STOP through the $AA- special message prefix. On receipt of the alert, the actual state of *xxxx*STOP is set to ACTIVE.

A similar method is used for the stopping operation.

Defining Resource Relationships

When you have defined the resources in a system image, you can specify the relationships between them. For each resource, you can specify two alternate relationships from it to other resources (see the section, *Primary and Alternate Relationships*, on page 5-48).

The region uses the relationship information during automated system startup and shutdown to determine the order in which to start or stop the defined resources.

You can define relationships between different types of resources within a system. The relationships between resources may be quite different for each system image definition held in the knowledge base.

There are two parties to a relationship: parent and child.

During automated operation, a parent must be active before dependent resources (its children) can be started. For example, JES must be active before CICS may be started. A child cannot be started automatically unless all its parents are active. For example, a DB2 resource may not be started unless CICS is active.

Similarly, a parent cannot be stopped automatically unless all its children are inactive.

You can use the GRT command to display resource relationships from the status monitor, the graphical monitor, and the System Image List panel.

Relate a resource to other resources as follows (see Figure 5-19):

Step 1. Enter **R** beside the resource in the resource list.

The Existing Relationships List panel lists any relationships already defined for the resource. Figure 5-19 shows that JES2 is defined as a parent of CICSD.

Step 2. Press F11 (Relate). A selection list of the resource classes is displayed. Select the class of resource to which you want CICSD to relate. For example, if you want to define a DB2 started task as a child of CICSD, enter **S** beside STC. A list of all started tasks in the system is displayed.

Step 3. Enter **C** beside DB2 to define it as a child of CICSD. The updated list of resource relationships is displayed.

Figure 5-19. Example—Relating Resources

```
SOLVPROD----- ResourceView : CICSD Existing Relationships List ---EASTTEST-0001
Command ===>                                                     Scroll ===> 10
             Press F11.
              R=Existing Related UP=Unlink Primary UA=Unlink Alternate U=Unlink
    Name               System Image Class  Primary Alternate
    JES2               EASTTEST0001 STC    PARENT
    CICSD              EASTTEST0001 STC    SELF    SELF
    **END**
```

```
SOLVPROD-------- Automation Services : Resource Class List --------------------
Command ===>                                                     Scroll ===> 10

                                              Use 'S' to select Class to process
    Class    Description
    DASD     Direct Access Device
    INIT     JES Initiator
    INTNL    ServiceView Internal
    JES      Job Entry Subsystem
    JOB      Batch Job
    LINE     JES Line
    PRT      Printer
    SPOOL    JES Spool
s   STC      Started Task
    SVC      Service
    TAPE     Tape or Cartridge
    USRCLS   User Class Resource
```

```
SOLVPROD---------- ResourceView : Relate Resources to CICSD -------EASTTEST-0001
Command ===>                                                     Scroll ===> 10

      P=Parent PP=Primary Parent AP=Alternate Parent UP=Unlink Primary U=Unlink
      C=Child  PC=Primary Child  AC=Alternate Child  UA=Unlink Alternate
    Name               System Image Class  Primary Alternate
    JES2               EASTTEST0001 STC    PARENT
    CICSD              EASTTEST0001 STC    SELF    SELF
    BACKUP             EASTTEST0001 STC
c   DB2                EASTTEST0001 STC
    DFHSM              EASTTEST0001 STC
```

Step 4. Press F3 (Exit) to return to the Existing Relationships List panel.

You can enter **R** beside a resource to display its other relationships. If you want to break a relationship with a resource, use the Unlink action codes.

Primary and Alternate Relationships

For each resource, you can define a primary relationship and an alternate relationship with other resources. During startup and normal operation, the region uses the primary relationship. During shutdown, the region uses the relationship specified in the OPSYSIDS parameter group. The default is the primary relationship.

To access the list of parameter groups, enter the **/ICS** shortcut.

Effect of Resource Relationships on Operations

Resource relationships affect systems operations in a number of ways, depending on the resource operation modes.

Starting a System That Contains Only Automated Resources

During system startup, the region starts all the resources without parents first, then the children, level by level, until all resources are started.

Resources are started subject to the availability requirements defined in the resource availability maps.

Stopping a System That Contains Only Automated Resources

During system shutdown, the region stops its resources in an order that is the reverse of the startup sequence. The resources are stopped level by level up the relationship tree.

Starting a System That Contains Automated and Manual Resources

The region starts resources in the same order as described in the section, *Starting a System That Contains Only Automated Resources*. However, when it encounters a resource in the MANUAL or IGNORED operation mode, the region does not start the resource or any of its children, even if the children are in the AUTOMATED operation mode. The region starts all other automated resources. Manual resources are started manually. Once a manual resource becomes active, the starting sequence continues with its children automatically.

Automated resources are started subject to the specified availability requirements.

Stopping a System That Contains Automated and Manual Resources

The region stops the resources in the same order as described in the section, *Stopping a System That Contains Only Automated Resources*. However, when it encounters a resource in the MANUAL or IGNORED operation mode, the region does not stop the resource or any of its parents, even if the parents are in the automated operation mode. The region stops all other automated resources. Manual resources are stopped manually. Once a manual resource becomes inactive, the stopping sequence continues with its parents.

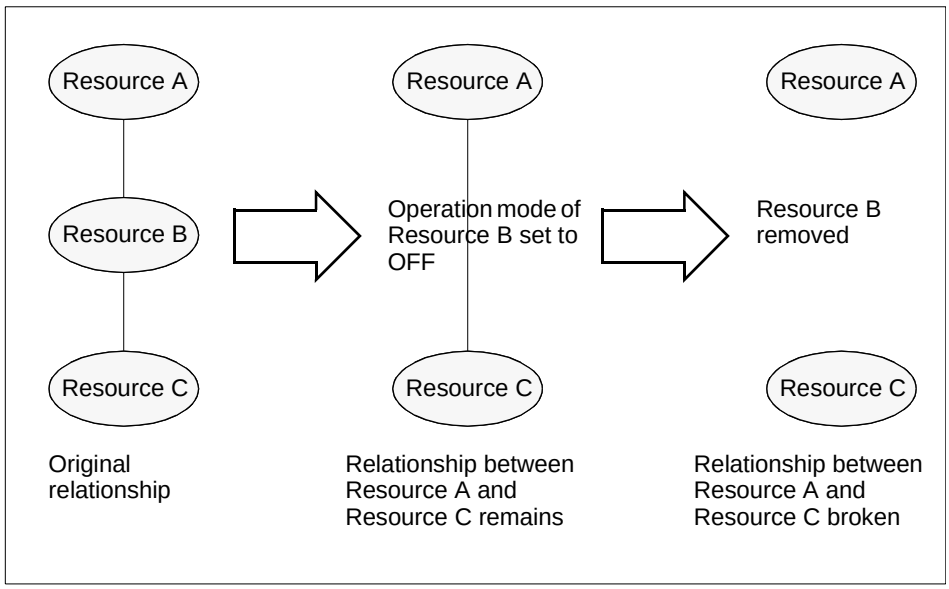
Note

The SHUTFORCE command enables an operator to stop all the resources defined in an active system image, irrespective of the operation mode. Before stopping the resources, the command sets all MANUAL or IGNORED operation modes to AUTOMATED operation modes. See the section, *Shutting Down All Resources*, on page 14-10 for information about the SHUTFORCE command.

Effect of a Resource Set to the OFF Operation Mode on a Relationship

Figure 5-20 shows the effect that a resource set to the OFF operation mode has on a relationship. As a comparison, the figure also shows the effect of removing a resource definition from the relationship.

Figure 5-20. Effect of a Resource Set to the OFF Operation Mode on a Relationship



Implementing Resource Definitions for System Load Balancing

In a multisystem environment, you might want to balance the load on the systems by moving the control of certain shared resources from one system to another. The MV command enables operators to perform that task.

During a move operation, the following actions occur:

1. The operation mode of the definition in the current system image is set to MANUAL.
2. The resource is stopped or inactivated in the current system.
3. The operation mode of the definition is set to OFF, removing the resource from the monitors. The control of the resource is removed from the current system.
4. The operation mode of the definition in the target system image is set to the value before Step 1. The status of the resource is redisplayed under the target system image. The control of the resource is moved to the target system.

For the command to be successful, you should define the shared resources to the affected system images as follows:

Note

If your product supports shared system images, you do not need to perform these steps. See your product user's guide.

Step 1. Define the required resources to the system image that normally controls those resources.

Step 2. Duplicate the relevant resource definitions, resource relationships, availability map, and processes to the system images that you want to include in the set of load sharing systems.

Note

The MV command uses the specified class and name in a definition to identify the resource. You do not need to duplicate the other information if you do not wish to.

Step 3. Set the operation mode of the duplicate resource definitions to OFF. That is, these definitions are initially dormant.

Step 4. If a resource is part of a service, specify all systems for that resource in the service definition. The service is then able to track that resource as it is moved from system to system.

Maintaining Resource Definitions

You can browse, update, copy, and delete resource definitions from the resource list.

Caution

Resources of the INTNL class with names in the form *xx*(*) are dynamic APPC resources that provide communications between regions. These resources are defined only for the duration of the APPC link.

Do *not* change these resource definitions.

Note

If you only want to hide a resource definition from the region, set the operation mode to OFF. The definition remains in the knowledge base but is not used.

When you copy a resource definition, the associated availability map and local processes are treated as follows if the map or process of the same name already exists in the target system image:

- For an availability map, you will be prompted to indicate whether you want to replace the existing map.
- For a local process, the existing process is *not* replaced.

6

Defining and Maintaining Services

A service is a collection of resources that support a business or operations function. Once you have defined the resources, you group relevant resources in service definitions. You use service definitions to specify the service availability requirements of your organization.

Note

Services are part of the automation feature. Some products do not use the automation feature. These products use Automation Services to provide only knowledge base management and presentation services.

This chapter contains the following topics:

- Accessing Service Definitions
- Actions That You Can Perform on a Service Definition
- Defining a Service
- Maintaining Service Definitions
- Backing Up Service Definitions

Note

You can define and manage services from focal point regions only. Services are not visible in subordinate regions, but you can include resources managed by a subordinate region in a service.

Accessing Service Definitions

Enter the **/SADMIN.S** shortcut to access the list of service definitions.

The Service List panel is displayed. The panel lists the services in the knowledge base.

Service definitions are stored in the knowledge base in a structure similar to that of resource definitions. Service definitions belong to the service system image, $SERVICE. Version 0001 of this image is always active. The definitions have a class of SVC.

Note

To assist with maintenance of your service definitions you can create backup versions of the $SERVICE 0001 service image. See *Backing Up Service Definitions*, on page 6-11 for details.

Actions That You Can Perform on a Service Definition

You can perform the following actions from the Service List panel:

F4 (Add)	Displays the Service General Description panel for you to define a new service definition. You cannot change the system name and version.
S, /, B (Browse)	Displays the Panel Display List panel that lists the service definition panels. When you select a panel, the panel displays the information in the BROWSE (read-only) mode. If you are authorized to update services, you can press F4 (Edit) from a definition panel to switch to the UPDATE mode.
U (Update)	Displays the Panel Display List panel that lists the service definition panels. When you select a panel, the panel displays the information in the UPDATE mode. You cannot change the system name, version, or service name.
C (Copy)	Displays the Service General Description panel in the COPY mode. You must change the service name.
D (Delete)	Displays a deletion confirmation message. Press ENTER to delete the service definition. Press F12 (Cancel) to retain the service definition.

Defining a Service

You can use variables as data in a service definition. See Table D-1 on page D-2 for the list of variables that you can use in a definition field.

To add a service definition, press F4 (Add) from the Service List panel. A Service General Description panel is displayed. You define the service by entering data on the following panels:

- Service General Description—You must complete this panel. The panel enables you to identify the service and to specify the service operation mode.
- Availability Map—The panel enables you to define the availability requirements for the service.
- Service Filters—You should complete this panel. The panel enables you to select members for the service and specify how important a member is to the service.
- State Thresholds—The panel enables you to define how the status of the service members affect the status of the service.
- State Change Exits—The panel enables you to specify state change exit processes that are invoked if the service changes to a given state.
- Automation Log Details—The panel enables you to change the logging requirements.
- Owner Details—The panel enables you to identify up to two people who can be contacted if the service has operational problems.
- Extended Function Exit—The panel enables you to specify an exit NCL procedure that can be used to extend the service functions provided in the region.

Describing the Service

Use the Service General Description panel to specify the service name, the operation mode, and a description of the purpose of the service.

Specifying the Operation Mode

Specify an operation mode of AUTOMATED, OFF, MANUAL, or IGNORED. During operation, the mode specified in the Operation Mode field can be restricted by the global operation mode. See Chapter 14, *Controlling the System Image*, for information about how to set the global operation mode.

Table 6-1 describes the effect of the operation mode on a service.

Table 6-1. Effect of Operation Mode on a Service

Operation Mode	Effect
AUTOMATED	The region monitors and automates the control of the service. When the desired state of the service is set to ACTIVE, the service places an ACTIVE desired state override on its members. The region then determines the actual state of the service from the actual states of the members. When the desired state of the service is set to INACTIVE, the service removes the ACTIVE desired state overrides from its members and acquires an INACTIVE actual state immediately.
OFF	The region neither monitors nor controls the service.
MANUAL or IGNORED	The region monitors but relinquishes control of the service to the operators. When the desired state of the service is set to ACTIVE, the service does not place the ACTIVE desired state overrides on its members. The overrides occur only when an operator starts the service manually by using the A(ctivate) command. Similarly, setting the desired state of the service to INACTIVE does not affect the members until an operator stops the service manually by using the T(erminate) command.

Defining the Availability of the Service

From the Service General Description panel, press F8 (Forward) to scroll forward to the Availability Map panel. Use this panel to define the changes to the normal availability of the service.

In a multisystem environment, you specify a system as the service automation focal point system and the scheduled times refer to the local times on that system. If the map schedules the starting of processes, the processes are started in the region on that system only.

Leave the Map Name field blank if you want to use the default desired state, which can be either ACTIVE or INACTIVE (as set in the AUTOIDS parameter group during region initialization).

The availability of a service overrides the availability of its members. If the service is always a member of another service, let the other service handle the availability of this service. Define the desired state of the service to be always inactive.

Note

You can create a new map from the service definition. You can name a new map and define it, or access an existing map, change the name, and update the copy. The map is created in the knowledge base when you save the definition.

Chapter 7, *Defining and Maintaining Availability Maps*, describes how to define availability maps.

Selecting Service Members

From the Availability Map panel, press F8 (Forward) to scroll forward to the Service Filters panel. Use this panel to define the filters that select the members of the service.

Define the filters by specifying the following criteria:

- The service class (SVC, if the member is another service) or resource class in the Class field.
- The name of the member in the Name field. You can use the following wildcard characters:
  - The underline character (_) represents a single character. For example, SOLV_SSI matches SOLV1SSI, SOLV2SSI, …
  - The percent character (%) represents zero or more characters. For example, SOLV%SSI matches SOLVSSI, SOLV1SSI, SOLV11SSI, …

    You can also use the asterisk (*) as a wildcard character. It behaves the same way as the % character except that you cannot have the * at the beginning of or embedded within the specified value. For example, * and SOLV* are valid values.
- The SMF ID of the system that owns the member in the SMF ID field. The default is the SMF ID of the local system.

  Note

  The SMF ID field specifies different values for VM and VOS3 systems. For the differences, see Help.

  When you have resources with the same identification defined on different systems and you want to include all those resources as members, specify ***** in the SMF ID field to indicate all systems.
- The type of resource (as specified in the resource definition) in the Type field. You can use the asterisk (*) wildcard character by itself or at the end of the specified value.

  Note

  The Type field is irrelevant for a service. Leave the value to the default.
- A weight that indicates how important the member is to the service in the Weight field.
- The type of weight in the Weight Type field.

You can define up to 97 lines of members.

Defining the Weight of a Service Member

The weight indicates how important a member is to the service. The valid values are 0% through 100%.

If the weight is 100%, the actual state of the member affects the actual state of the service directly. For example, if the member fails, the service fails.

If the weight is 0%, the member has no effect on the service.

If the weight is between 0% and 100%, the effect of the member on the service depends on the state thresholds. The section, *Defining the State Thresholds*, on page 6-8, describes the state thresholds.

You can apply the following two types of weights to service members: fixed and proportional.

Using a Fixed Weight

With a fixed weight, every member included in a line entry has the weight specified in the Weight field.

In the following two examples, the weight is 100% fixed:

- If the line entry includes only one member (for example, the CICSD started task in the EASTTEST 0001 system), the member has 100% weighting in the service.
- If the line entry includes more than one member (for example, the CICSD started tasks in all the connected systems (SMF ID=*)), each of the members has 100% weighting in the service.

Using a Proportional Weight

You can use the proportional type of weight when the line entry includes more than one member. With a proportional weight, every member included in the line entry has an equal proportion of the weight specified in the Weight field.

For example, if the weight is 100% proportionally applied to two members, each member has 50% weighting in the service.

Viewing the Service as Defined by the Service Filters

The service filters select the members for a service. Only members defined in the currently active system images are selected. Therefore, the members can change if the active system images change (for example, when a connected region has a different system image loaded).

You can use the F5 (Model) function key to view the members in the service.

Defining the State Thresholds

From the Service Filters panel, press F8 (Forward) to scroll forward to the State Thresholds panel. Use this panel to define how the actual states of the members affect the actual state of the service.

The actual state of a service can be one of the following: UNKNOWN, FAILED, ACTIVE, STARTING, and DEGRADED. Assign a threshold to the first four states.

The state threshold is set by the specified combined weight of the service members that have particular actual states. The region checks the states in the sequence UNKNOWN to STARTING until a threshold is found to be equalled or exceeded. That is, if the UNKNOWN state threshold requirement is satisfied, the service takes on the UNKNOWN state irrespective of whether the other threshold requirements are satisfied. If none of the threshold requirements are satisfied, the service takes on the DEGRADED state.

If a service filter finds no members, its weight specified in the Weight column on the Service Filters panel is added to the combined weight for the UNKNOWN state.

Example

You provide a printing service by using 10 printers. The service agreement requires that at least two printers must be available 24 hours per day. The service is active if all printers are available. The service is degraded if at least two but not all printers are available. The service fails if less than two printers are available. You define a 100% proportional weighting distributed across the printers. That is, each printer has a weight of 10%.

You define the state thresholds as follows:

Actual State	**Threshold**
UNKNOWN	100%
FAILED	90%
ACTIVE	100%
STARTING	10%

During operation, the service takes on the actual states as follows:

Actual State of Members	**Actual State of Service**
All printers are in the UNKNOWN state.	UNKNOWN
Nine or more printers are in the DEGRADED, FAILED, INACTIVE, or STOPPING state.	FAILED
All printers are in the ACTIVE state.	ACTIVE
One or more printers are in the STARTING state.	STARTING
The printer states do not satisfy one of the above conditions.	DEGRADED

Implementing the State Change Exits

From the State Thresholds panel, press F8 (Forward) to scroll forward to the State Change Exits panel. This panel enables you to specify two types of exit processes:

- A process that executes before the service is started. By using this feature, you can add your own pre-activation tasks to the internal service starting method.
- Processes that execute on specified state changes. For example, if a service fails, you may want to invoke a procedure that writes a problem report. You can specify a process to execute on changes to the actual state, the desired state, or the logical state of the service.

In a multisystem environment, you can specify whether the processes are executed in a specific region only or in all connected regions.

Defining the Logging Details

From the State Change Exits panel, press F8 (Forward) to scroll forward to the Automation Log Details Panel. This panel contains information about the size of the temporary log for the service (called a *transient log*), the destination of the logged information, and the type of information logged.

Specifying the Owner Details

From the Automation Log Details panel, press F8 (Forward) to scroll forward to the Owner Details panel. This panel enables you to identify up to two people who can be contacted if this service has operational problems.

Implementing the Extended Function Exit

From the Owner Details panel, press F8 (Forward) to scroll forward to the Extended Function Exit panel. The panel enables you to provide additional operator functions. Specify the exit NCL procedure that provides these functions. The procedure is invoked when an operator issues the XF command against the service.

The extended function exit NCL procedure has access to variables with the prefix ZRM. Appendix D, *Variables*, describes these variables.

Maintaining Service Definitions

You can browse, update, copy, and delete service definitions from the Service List panel.

Note

If you only want to hide a service definition from the region, set the operation mode to OFF. The definition remains in the knowledge base but is not used.

Backing Up Service Definitions

To assist you with the maintenance of your service definitions, you can create backup versions of your service image. By creating a backup version of your service image or definitions, you can perform the following:

- Update service definitions in any version of a service image
- Restore a service definition from a backup service image
- Merge two versions of a service image

To create a backup version of a service image, perform the following steps:

Step 1. Enter the **/SADMIN.SI** path to access the service image list (see Figure 6-1). If you have not created a backup before, there is only one service image listed: $SERVICE 0001.

Note

The active service image can only be $SERVICE 0001. $SERVICE 0001 cannot be deleted.

Step 2. Enter **C** beside the service image you want to copy. The ServiceView : Service Image Definition panel is displayed (see Figure 6-2).

Figure 6-1. Copying a Service Image

```
SOLVPROD------------ ServiceView : Service Image List ------------------------
Command ===>                                                   Scroll ===> PAGE

                            S/B=Browse U=Update C=Copy D=Delete L=List Services
    Sysname   Ver    Short Description
 C  $SERVICE  0001   Base $SERVICE 0001 Image
```

Step 3. Enter a new Database Version, Short Description, and Long Description on the Service Image Definition Panel.

Figure 6-2. Creating a Backup Service Image

```
SOLVPROD----------- ServiceView : Service Image Definition --------------------
Command ===>                                                    Function=COPY


System Name ......... $SERVICE
Database Version .... 0002


Short Description ... Backup Service Image______
Long Description .... Service Image used for backups___________________________
                      ______________________________________________________________
                      ______________________________________________________________
                      ______________________________________________________________
```

Step 4. Press F3 (File) to save the backup version of the service image. A copy in progress panel is displayed while the copy occurs. You are then returned to the Service Image List with the backup version displayed in the list.

Updating Service Definitions in a Backup Service Image

You can access a list of all the service definitions in any version of a service image. From this list you can update any service definitions contained in the service image.

To update a service definition in the $SERVICE 0002 backup image created above, perform the following steps:

Step 1. Enter **L** (List Services) beside the $SERVICE 0002 service image in the Service Image List. The ServiceView : Service List panel is displayed showing all the service definitions in that service image.

Note

You can access the list of service definitions for another version of the service image by changing the version number on the Service Image List panel and pressing ENTER.

Step 2. Enter **U** beside the service definition that you want to update. The ServiceView : Panel Display List is displayed for that service definition.

Step 3. Update the service definition as required. See the information at the beginning of this chapter for details on how to update a service definition.

Step 4. Press F3 (File) to save the changes. You are returned to the ServiceView : Service List panel.

Restoring a Service Definition from a Backup Service Image

If you have made changes to a service definition and you are having trouble with its implementation, you can restore it from a previous version of the definition.

To restore service definition SERV01from $SERVICE 0002 service image to $SERVICE 0001, perform the following steps:

Step 1. Enter **C** beside SERV01 in the ServiceView : Service List. The ServiceView : Service Image Definition panel is displayed (see Figure 6-3)

Figure 6-3. Restoring a Service Definition

```
SOLVPROD--------- ServiceView : Service General Description ------$SERVICE-0002
Command ===>                                                      Function=COPY

System Name ..+ $SERVICE   Database Version ..+ 0002


Service Name ......... SERV01______



Operation Mode .....+ MANUAL_____
Short Description ... Service One ___________________
Long Description .... Service definition number one_____________________________
                      ______________________________________________________________
                      ______________________________________________________________
                      ______________________________________________________________
```

Step 2. Change the Database Version from 0002 to 0001 and press F3 (File) to save the change. As there is already a copy of the service in the target service image, the Confirm Copy Replace panel is displayed (see Figure 6-4).

Figure 6-4. Confirm Copy Replace Panel

```
SOLVPROD------- Automation Services : Confirm Copy Replace -----------------FTI


. Note ---------------------------------------------------------------------.
| This resource definition already exists in the target system image.        |
| If you continue this copy process with replace, the resource definition in  |
| the target system image will be replaced but any existing relationships are |
| preserved.                                                                 |
'----------------------------------------------------------------------------'

    Press ENTER to Confirm Copy Replace, Cancel Key to cancel copy process


            . Resource to Copy ---------------------------------------.
            |                    From              To                |
            | System Name ...... $SERVICE          $SERVICE          |
            | System Version ... 0002              0001              |
            | Resource Class ... SVC               SVC               |
            | Resource Name .... SERV01            SERV01            |
            '-------------------------------------------------------'


 F1=Help      F2=Split
                          F9=Swap                                F12=Cancel
```

Step 3. Press ENTER to confirm the replace or F12 (Cancel) to cancel the request. You are returned to the ServiceView : Service List panel.

Merging Two Service Images

You can merge two service images and replace the active system image with a backup version.

To merge service images $SERVICE 0002 and $SERVICE 0001, complete the following steps:

Step 1. Enter **C** beside the $SERVICE 0002 on the ServiceView : Service Image List panel. The ServiceView : Service Image Definition panel is displayed.

Step 2. Change the Database Version number to 0001and press F3 (File). The Confirm System Image Merge panel is displayed (see Figure 6-5)

Figure 6-5. Confirm System Image Merge Panel

```
SOLVPROD----- Automation Services : Confirm System Image Merge  ---------------


                            MERGE TWO SYSTEM IMAGES

     .-------------------------------------------------------------------.
     |  WARNING: THE SPECIFIED TARGET SYSTEM IMAGE ALREADY EXISTS.        |
     |           THIS REQUEST WILL MERGE ALL THE COMPONENTS FROM THE      |
     |           SOURCE SYSTEM IMAGE INTO THE TARGET SYSTEM IMAGE.        |
     |                                                                    |
     |              * Source System Image - $SERVICE-0002                 |
     |              * Target System Image - $SERVICE-0001                 |
     |                                                                    |
     |    Enter 'YES' to OVERLAY like-named components .... NO_           |
     |                                                                    |
     '-------------------------------------------------------------------'

     Press Confirm key to confirm request, Cancel key to cancel the request



 F1=Help      F2=Split                                            F6=Confirm
                           F9=Swap                              F12=Cancel
```

Step 3. Enter **YES** in the input field if you want to overlay like-named components.

Step 4. Press F6 (Confirm) to confirm the merge, or F12 (Cancel) to cancel the merge.

7

Defining and Maintaining Availability Maps

An availability map enables you to define the availability requirements for a service or resource. An availability map also enables you to schedule the execution of processes. You can add an availability map at any time. The map becomes effective as soon as you attach services or resources to it.

Note

Availability maps are part of the automation feature. Some products do not use the automation feature. These products use Automation Services to provide only knowledge base management and presentation services.

This chapter contains the following topics:

- Implementing Availability Maps
- Accessing Availability Map Definitions
- Actions That You Can Perform on an Availability Map Definition
- Defining an Availability Map
- Viewing Timer Information
- Attaching a Service or Resource Definition to an Availability Map
- Detaching Service or Resource Definitions From an Availability Map
- Maintaining Availability Map Definitions

Implementing Availability Maps

The status of a service or a resource is determined by the desired state information specified in its definition. The definition can include an availability map that schedules changes to the default availability. These changes are activated by timers.

Note

The default availability of a service or resource is determined by the default desired state, which is set in the AUTOIDS parameter group during region initialization. The Customization Parameters panel lists the region parameter groups. Enter the **/ICS** shortcut to access the panel.

Availability maps enable you to schedule changes to the default availability requirements of one or more services or resources. The service image and each system image has its own set of availability maps. You define an availability map (for example, MAP1) and attach as many services or resources to the map as required. Because availability maps are not limited to a seven-day cycle, you can define changes to the availability requirements that apply daily, on the same day every week, on the same date every month, for a specific date and time, and so on. You can also suppress changes temporarily and update timer information at any time.

An availability map has two parts: a map definition and a timer definition. The map definition contains information about the map itself. The timer definition contains information about when to change the desired state of the services or resources that use this map. The timer definition can also contain information about when to change the operation mode and when to start processes to perform special tasks.

There are two stages in creating an availability map:

1. Creating an availability map—use the Availability Maps option on the Definition Menu to create a map definition and to define the timer information for the map.

   Note

   You can also create an availability map while you are defining a service or resource.

   When you have saved the map definition, you can attach services or resources to that map.

2. Attaching services or resources to a map—use the Services or the Resources option on the Definition Menu to attach individual services or resources to a map.

Rules for Availability Map Definitions

The following rules apply to availability maps:

- If the timer definition is blank, it means that default availability requirements apply to all the services or resources attached to that map.
- A map only applies to the service image or the system image for which it is defined.
- Map names must be unique in the image to which the map applies.

Accessing Availability Map Definitions

Define as many maps for a system image as you require. Once the map is defined, you can define timer information and attach services or resources to the map. Use the Availability Maps option to create and maintain availability map definitions.

The service image and each system image has its own set of availability maps. Access the availability map definitions as follows:

- Enter the **/SADMIN.A** path to access the list of service availability maps.
- Enter the **/RADMIN** shortcut to access the Resource Administration menu. Type the option code **A**, and the name and version of the system image that owns the maps you want to create or access, and press ENTER.

The Availability Map List panel is displayed. This panel lists the availability maps for the specified service or system image.

Temporary Availability Maps

A temporary availability map is an availability map created from the status monitor to override the current map attached to a service or resource. A temporary map has an expiry time when the map is deleted automatically. You can use a temporary map as any other map, remembering that it has a defined life time.

Actions That You Can Perform on an Availability Map Definition

You can perform the following actions from the Availability Map List panel:

F4 (Add)	Displays the Availability Map panel for you to define a new map definition.
S, /, B (Browse)	Displays the map and timer definition information in the BROWSE (read-only) mode. If no times are defined, the timer definition box is blank. If you are authorized to update maps, you can press F4 (Edit) to switch to the UPDATE mode.
U (Update)	Displays the map and timer definition information in the UPDATE mode. You cannot change the system name, system version, or map name.
C (Copy)	Displays the map and timer definition information in the COPY mode. You must change at least the system name, system version, or map name. When you use this action, all timer definitions are also copied.
D (Delete)	Displays a deletion confirmation panel. The panel lists the services or resources attached to the map (if applicable). Press ENTER to delete the map, including the timer definition information and the links to services or resources. Press F12 (Cancel) to retain the map definition. You can use the LR action to list any attached services or resources. Deleting a map with attached services or resources removes the links and causes them to acquire the default availability, which is dependent on the default desired state set in the AUTOIDS parameter group during region initialization.
LR (ListAttRes)	Displays a list of services or resources attached to a map.
N (NextTimers)	Displays the Next Timers Execution Time panel. This panel lists the upcoming changes in the availability requirements defined in a map, in chronological order from next to last.

Defining an Availability Map

From the Availability Map List panel, press F4 (Add) to add an availability map. An Availability Map panel is displayed. On this panel, specify the timer information that sets the availability requirements.

Defining Timers

You can define two types of timer information:

- For all services or resources—define the timer, leaving the SVC/Resource Name field blank. This timer information applies to any services or resources attached to the map.
- For a specific service or resource—define the timer with the name of the service or resource in the SVC/Resource Name field. This timer information applies to the named service or resource if the service or resource is attached to the map.

You can use the action codes to repeat or delete rows of information, or to insert blank lines.

Use the following values in the Day field to simplify data entry:

Value	Function
*	Repeats the timer for all days (that is, Monday through Sunday).
W/D	Repeats the timer for weekdays (that is, Monday through Friday).
W/E	Repeats the timer for weekends (that is, Saturday and Sunday).

Leave the Day field blank if you fill in the Date field. If the Mode field is left blank, you will not override the operation mode.

Scheduling Processes

If you want the map to start processes at defined times, press F11 (Right) to display the fields for specifying processes.

Resetting Manual Overrides

You can use a timer to reset manual desired state and operation mode overrides. Specify RESET in the Des.State and in the Mode fields.

When a manual override exists, the scheduled change to the overridden parameter cannot be made. If you want to ensure that the scheduled changes are made, reset the overrides first.

See the section, *Overriding the Desired State*, on page 13-19 for information about how a desired state can be overridden manually. See the section, *Overriding the Operation Mode*, on page 13-18 for information about how an operation mode can be overridden manually.

Example

Note

The following example, though describing how to define an availability map for services, is also applicable to defining an availability map for a system image.

In this example, you define a map for the defined services to schedule such things as availability during holidays and when system maintenance is required. The map is called MAP1 and uses SOLV as the service automation focal point system.

Use the **/SADMIN.A** path and the F4 (Add) function key to access the Availability Map panel. On the panel, you type **MAP1** in the Name field, a description in the Description field, and **N** in the Expire Delete field to retain expired timer events. (These events occur on specific dates.) You can now specify timer details.

You want to stop all services on 25 May 1997 at 0830 hours for system maintenance. All services are to be reactivated at 1600 hours on the same day. (Resources that belong to the services should have a scheduled INACTIVE desired state for all times. That is, the services control the availability of those resources by using the ACTIVE desired state overrides.)

In the Timer Details box, type the information about the date, the time, the change to the status, and whether this change is to be processed. In order to have a change processed, specify **ON** in the Status column. Figure 7-1 shows the completed Availability Map panel.

Figure 7-1. Example—Availability Map

```
SOLVPROD---------- Automation Services : Availability Map ---------Function=ADD
Command ===>                                                  Scroll ===> 10

+ Availability Map -------------------------------------------------------+
| System Name .. $SERVICE  Version .. 0001   Last Updated By               |
| Name ......... MAP1                        at           on               |
| Expire Delete  NO        Description ... Changes to service availability |
| Timer Execution Control System ........+ SOLV     (Service/Shared Images) |
| Attached Resources ... _                                                 |
|--------------------------------------------------------------------------+
+ Timer Details ----------------------------------------------------------+
|                                                 D=Delete I=Insert R=Repeat |
|     Day Date        Time     SVC/Resource Name  Des.State Mode       Status |
| ___ SUN 25-MAY-1997 08.30.00 __________________ INACTIVE  __________ ON     |
| ___ SUN 25-MAY-1997 16.00.00 __________________ ACTIVE    __________ ON     |
| ___ ___ ___________ ________ __________________ _________ __________ ___    |
| ___ ___ ___________ ________ __________________ _________ __________ ___    |
| ___ ___ ___________ ________ __________________ _________ __________ ___    |
| ___ ___ ___________ ________ __________________ _________ __________ ___    |
| ___ ___ ___________ ________ __________________ _________ __________ ___    |
| ___ ___ ___________ ________ __________________ _________ __________ ___    |
|  F1=Help      F2=Split     F3=File      F4=Save      F5=NextTmr   F6=Sort   |
|  F7=Backward  F8=Forward   F9=Swap                   F11=Right    F12=Cancel |
+-----------------------------------------------------------------------------
```

Viewing Timer Information

The Next Timers Execution Time panel lists information about upcoming changes to availability. You can obtain two different views of this timer information by:

- Viewing the timer information in all availability maps for the services or in a system image
- Viewing the timer information in one availability map

The views list the next invocation of the defined timers. For example, a timer that executes every Monday is listed once only.

Viewing All Timer Information

You can view a list of the upcoming changes scheduled in all maps defined in the service image or in a system image. The changes are listed in chronological order.

Access this information from the Availability Maps List panel by pressing F12 (NextTmrs). This displays a Next Execution Time panel, which lists the upcoming changes for all the maps.

Viewing the Timer Information in One Availability Map

You can view a list of the upcoming changes scheduled in an individual map. The changes are listed in chronological order.

You can view the timer information in a map from the following panels:

- From the Availability Map List panel, enter **N** next to an availability map to select the NextTimers action.
- From an Availability Map panel (while you are working on an availability map definition, a resource definition, or a service definition), press F5 (NextTmr).

A Next Execution Time panel is displayed, listing the upcoming changes for the selected map.

Attaching a Service or Resource Definition to an Availability Map

Once a map is defined, you can attach service or resource definitions to it, using the Services or the Resources option on the appropriate Definition Menu to access the Availability Map panel. On the Availability Map panel, enter the name of the required map in the Map Name field to attach the definition to the map. This panel also enables you to add, update, or copy timer information that is specific to the service or the resource. Timer information in the map is available to the attached services and resources immediately.

Example

> **Note**
> The following example, though describing how to attach a service to an availability map, is also applicable to attaching a resource to an availability map.

In this example, you are attaching a credit authorization service (CREDITAUTH) to MAP1. The CREDITAUTH service is in the MANUAL operation mode, and you want to it to change to the AUTOMATED operation mode on Tuesday, 20 May 1997.

The following actions are required to attach the service and to define the timer information for it:

Step 1. Enter the **/SADMIN.S** path to access the list of services.

Step 2. Enter **U** next to the CREDITAUTH service definition to display a Panel Display List panel. This panel lists all the service definition panels. You can update any of these panels.

Step 3. Enter **S** next to SVC CREDITAUTH Availability Map to select the availability map panel. An Availability Map panel is displayed.

Step 4. Type the name of the map the service is to use, and press F10 (EditMap) to edit the timer information. During editing, you can copy and delete the existing timer information or insert extra lines to add new timer information.

Step 5. Press F3 (OK) to finish the edit. Figure 7-2 shows the completed panel.

Figure 7-2. Example—Service Definition Availability Map Panel

```
SOLVPROD------ ServiceView : SVC CREDITAUTH Availability Map -----$SERVICE-0001
Command ===>                                                     Function=UPDATE

+- Map Details ---------------------------------------------------------------+
| Map Name .....+ MAP1________                          Expire Delete ... NO    |
| Description ... Changes to service availability                               |
| Timer Execution Control System .........+ SOLV      (Service/Shared Images) |
| Attached Resources ...+ _                                                     |
+-------------------------------------------------------------------------------+
+- Timer Details -------------------------------------------------------------+
|     Day Date        Time     SVC/Resource Name  Des.State Mode       Status |
|     TUE 20-MAY-1997 08.00.00 CREDITAUTH                   AUTOMATED  ON     |
|     SUN 25-MAY-1997 08.30.00                     INACTIVE             ON     |
|     SUN 25-MAY-1997 16.00.00                     ACTIVE               ON     |
```

Detaching Service or Resource Definitions From an Availability Map

You can detach service or resource definitions from an availability map (for example, if you want to make changes to a resource and test it separately).

You can detach a service or a resource from an availability map in two ways:

- By using the list of attached services or resources
- By using the service or resource definition panels

Detaching Services or Resources From a List

From any panel, proceed as follows:

Step 1. Use the **/SADMIN.A** (for services) or the **/RADMIN.A** (for resources) path to access the Availability Map List panel.

Step 2. Enter **LR** next to the map from which you want to detach services or resources. A list of the attached services or resources is displayed.

Step 3. Enter **DT** next to the appropriate services or resources to detach them from the map.

Detaching a Service or Resource While You Are Working on Its Definition

You can detach a service or a resource from a map while you are working on the definition. Proceed as follows:

Step 1. Select the Availability Map panel to display the availability map to which the definition is attached.

Step 2. Position the cursor in the Availability Map Name field and clear the field to detach the service or resource.

You can attach a service or resource to a different map, by typing over the name in the Map Name field and saving the definition.

Maintaining Availability Map Definitions

You can browse, update, copy, and delete timer information and availability map definitions from the Availability Map List panel.

8

Defining Processes to Implement Complex Operations

A process is a series of steps that can be executed in sequence to perform complex processing.

This chapter contains the following topics:

- Implementing Processes
- Accessing Process Definitions
- Actions That You Can Perform on a Process Definition
- Defining a Process
- Testing a Process
- Logging Process Activities
- Maintaining Process Definitions
- Backing Up Global Processes

Implementing Processes

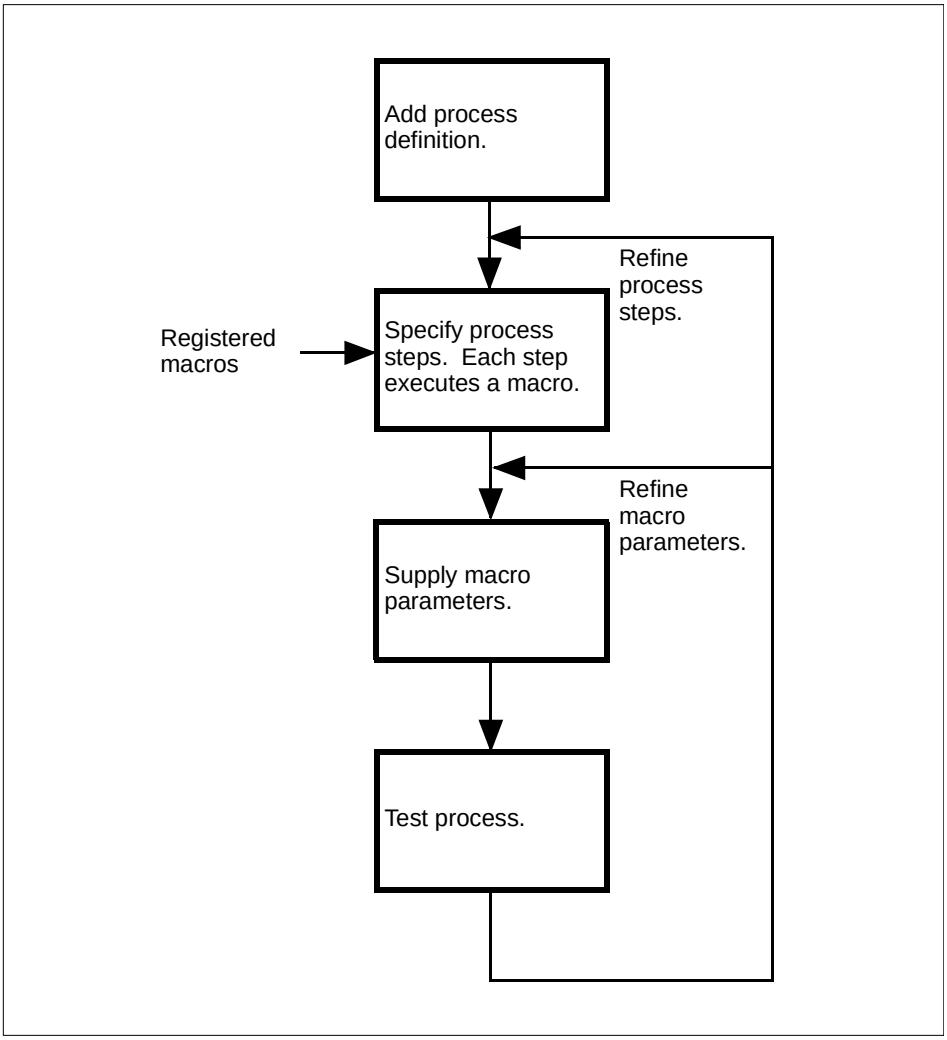
You define processes to automate complex operations tasks. For example, before you stop a CICS resource, you broadcast a warning to the current CICS users, and check the number of users and CICS transactions that still require that resource. When all users have exited CICS, you cancel any active transactions and stop CICS. These actions can be specified in a process definition and run as a single task.

Processes can be executed as follows:

- From a resource definition—you can specify a process in a resource definition (for example, where the resource stopping method is complex). The process is invoked when required for that resource.
- From an availability map—you can specify a process in an availability map (for example, to perform tasks at particular times).
- From an event rule—you can specify a process in an event rule. The process is invoked when an event triggers the rule.
- As a single task—you can run a process as a single, independent task. Use this feature to debug processes or as a quick way of executing a process manually.
- Interactively—you can run a process in the INTERACTIVE mode. Use this feature to check the results of processing single steps, or of processing a sequence of steps one at a time. You can display individual step logs and, if required, change the step parameters.

Figure 8-1 shows the typical stages in defining a process.

Figure 8-1. Defining a Process



Process Types

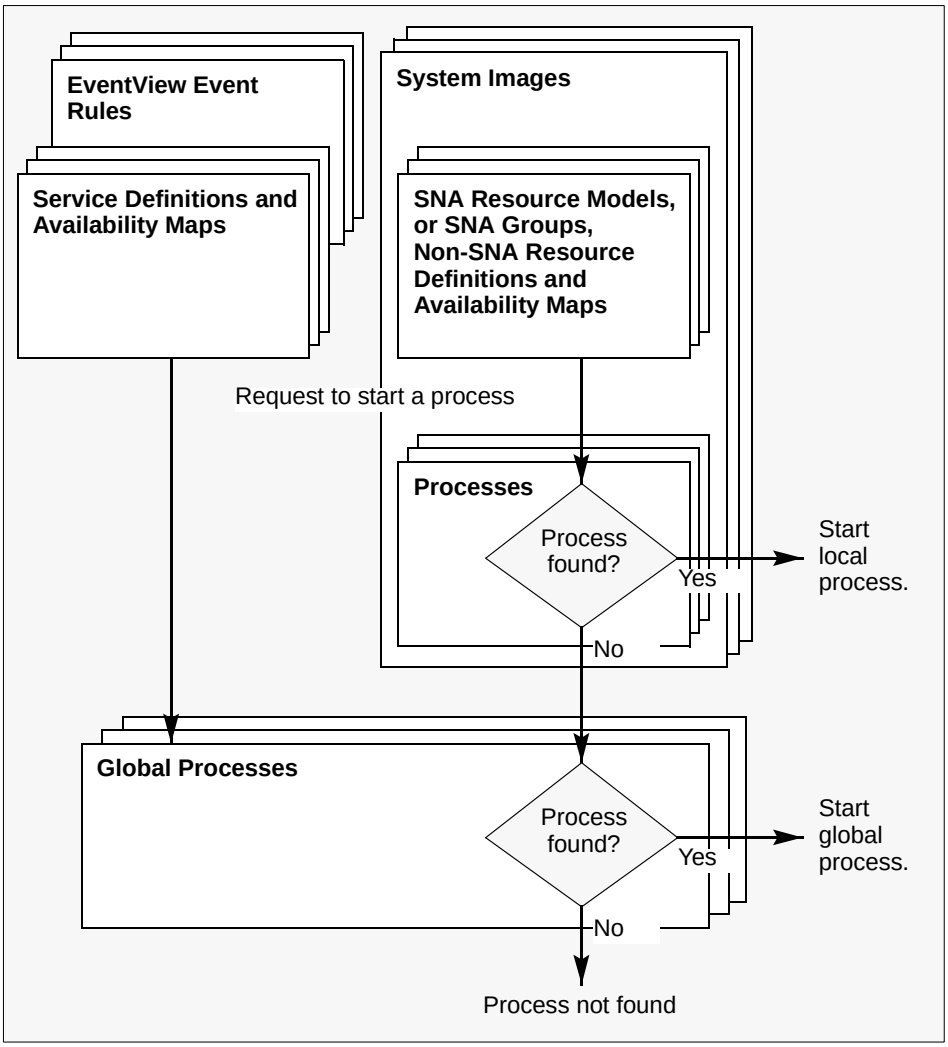
A process can be global (available to all components) or local (available to a specific system image only). A global process is available for use by all components while a local process is available for use only if it belongs to the local active image.

ServiceView and EventView components can use global processes only. ResourceView components can use both types of processes, according to the following rules:

1. If a process is required and one exists in the local active system image, that process is used.
2. If the required process does not exist in the active system image, the global process of the same name is used.

Figure 8-2 shows how processes are searched for execution.

Figure 8-2. Process Search Path



Accessing Process Definitions

Each system image has its own set of processes. Access the process definitions as follows:

- Enter the **/RADMIN** shortcut to access the Resource Administration menu. Type the option code **P**, and the name and version of the system image that owns the processes you want to create or access, and press ENTER.

  The Process List panel is displayed. This panel lists not only the processes in the system image but also the global processes (displayed in blue on a color terminal).

- Enter the **/RADMIN.GP** or the **/SADMIN.GP** path to access the list of global processes. Global processes belong to the $PROCESS 0001 system image.

  The Process List panel is displayed. This panel lists the global processes.

Actions That You Can Perform on a Process Definition

You can perform the following actions from the Process List panel:

F4 (Add)	Displays the Process Definition panel for you to define a new process definition to the system image.
S, /, B (Browse)	Displays the process definition information in the BROWSE (read-only) mode. If you are authorized to update processes, you can press F4 (Edit) to switch to the UPDATE mode.
U (Update)	Displays the process definition information in the UPDATE mode. You cannot change the system name, system version, or process name.
C (Copy)	Displays the process definition information in the COPY mode. You must change at least the system name, system version, or process name.
D (Delete)	Displays a deletion confirmation message. Press ENTER to delete the process definition. Press F12 (Cancel) to retain the process definition.
E (Exec Process)	Executes the process.
I (Interactive Mode)	Displays the process definition information in the INTERACTIVE mode. In this mode, you can execute the individual steps in the process.

Defining a Process

From the Process List panel, press F4 (Add) to add a process definition. A Process Definition panel is displayed.

To define a process, first decide what you want the process to do, then break it down into steps, each step representing an action. Specify a macro for each step. A macro is an NCL procedure that performs the processing for that step. Sample macros are listed in Table E-1 on page E-2. Authorized users can use the Register Macros option to register new macros. (The *Automation Services Administrator Guide* describes how to register macros.)

Step processing may be conditional on the processing result of an earlier step. In the example below, STEP2 runs if STEP1 processing returns a code of 0. STEP3 runs if STEP1 processing returns a code greater than 0.

StepName	**Step/RC**	**Opr**	**R/C**
STEP1			
STEP2	STEP1	=	0
STEP3	STEP1	>	0

When you define a process on the Process Definition panel, you must complete the Name and Description fields to identify the process, and the StepName and Macro fields to define each step. If you want to find out what macros are available, enter **?** in a Macro field to display the list of available macros.

Caution

$NCL is the name of a special process definition. Do *not* use this name when you add process definitions.

Conditions are optional. Use relational operators in the Opr fields to set the conditions. Enter **?** in an Opr field to identify the valid relational operators. The section, *Relational Operators*, on page G-2 describes the relational operators.

You can repeat and delete steps, and insert blank lines.

The return code from a process is the return code from the last executed process step.

Figure 8-3 shows an example of a process that issues multiple system commands to stop a resource called CA7.

Figure 8-3. Example—Process

```
SOLVPROD--------- Automation Services : Process Definition --------Function=Add
Command ===>                                                   Scroll ===> 10

+ Process Definition -------------------------------------------------------+
| System Name .. SOLV      Version .. 0001   Last Updated By                 |
| Name ......... SHUTCA7                     at           on                 |
| Description .. SHUT DOWN CA7                                              |
+---------------------------------------------------------------------------+
+ Process Steps -------------------------------------------------------------+
|                                                                           |
|                                          D=Delete I=Insert P=Parms R=Repeat |
|                     Condition                                             |
|      StepName   Step/RC  Opr R/C   Macro    Description                   |
|      STEP1                         SYSCMD   EXECUTE A COMMAND             |
|      STEP2      STEP1    =   0     SYSCMD   EXECUTE A COMMAND             |
|      STEP3      STEP2    =   0     SYSCMD   EXECUTE A COMMAND             |
|      STEP4      STEP1    =   99    SYSCMD   EXECUTE A COMMAND             |
|                                                                           |
|                                                                           |
|                                                                           |
|                                                                           |
|  F1=Help     F2=Split    F3=File     F4=Save                              |
|  F7=Bkwd     F8=Forward  F9=Swap                F11=Right   F12=Cancel   |
+---------------------------------------------------------------------------
```

If STEP1 completes successfully, STEP2 executes the next shutdown command. If STEP2 completes successfully, STEP3 issues the final shutdown command.

If STEP1 fails, STEP4 executes and issues a CANCEL command.

Setting Macro Parameters

When you select a macro, it contains either no parameters or default parameters. To set the parameters for a macro, enter **P** beside the process step. A Macro Parameter Definition panel is displayed. The parameters required by each macro depend on the purpose of the macro. Change the parameters as required, and press F3 (OK).

Figure 8-4 shows the parameters set for STEP1 in the previous example shown in Figure 8-3.

Figure 8-4. The Parameters for STEP1

```
SOLVPROD--- Automation Services : SYSCMD Macro Parameter Definition -----------
Command ===>                                                   Function=UPDATE

+- System Command -------------------------------------------------------------+
|                                                                              |
| Command ..... F CA7T,/LOGON MASTER_____________________________________ |
|                                                                              |
| Jobname ..... ________                                                       |
|                                                                              |
| Wait Time ... 30__  Wait Time Expiry Return Code ... 16_                     |
+------------------------------------------------------------------------------+
+- Response Message Analysis ---------------------------------------------------+
|                                D=Delete Extended Filter  S=Extended Filter |
|           Message Text                                     Return Extended |
|                                                            Code   Filter?  |
|   ___     CA-7.023 - V3.0 (9106) OPERATOR IS LOGGED ON_    0__    NO       |
|   ___     ______________________________________________   ___             |
|   ___     ______________________________________________   ___             |
|   ___     ______________________________________________   ___             |
|   ___     ______________________________________________   ___             |
|                                                                              |
+------------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=OK
                           F9=Swap                                F12=Cancel
```

The parameters include:

- The system command issued
- The text of the expected response
- A processing return code of 0
- A wait time of 30 seconds
- A timeout return code of 16

You may also choose to specify an extended filter for the analysis of the response message text. The filter is similar to the filter described in the section, *Defining an Extended Message Filter*, on page 5-36. For example, a response might contain variable information, and you might want to accept the message only if it contains specific values.

Caution

You *cannot* use the special message prefixes described in the section, *Using Special Message Prefixes*, on page 5-35 in process definitions.

Using a Variable as a Macro Parameter

You can use a variable to hold the value of a macro parameter. You pass the value of any variables required by a process as parameters when you specify the process, for example, in a resource definition. Do *not* specify variable names that start with #, $, or Z. These names are reserved for use by the Automation Services software.

Example

You have defined a process that contains the SYSCMD macro which issues the $DU,&PRT command. When you use the process, you supply the value of the &PRT variable by specifying the following parameter: PRT=*printer-name*. Specify the name of the variable only (without the &).

Example—Defining a Display Process

Figure 8-5 shows the steps of a sample process that displays information about a printer.

Figure 8-5. Example—Printer Display Process Steps

```
SOLVPROD--------- Automation Services : Process Definition -----Function=Browse
Command ===>                                                   Scroll ===> HALF

                                                                          P=Parms
                    Condition
     StepName   Step/RC  Opr R/C   Macro    Description
     DISPPRTR                      SYSCMD   DISPLAY PRINTER
     STOPPING   DISPPRTR EQ  2     SETSTATE ACTSTAT=STOPPING. CLEAR EXTDISP
     INACTIVE   DISPPRTR EQ  5     SETSTATE ACTSTAT=INACTIVE. CLEAR EXTDISP
     ACTIVE     DISPPRTR EQ  4     SETSTATE ACTSTAT=ACTIVE.   CLEAR EXTDISP
     PRINTING   DISPPRTR EQ  0     SYSCMD   DISPLAY PRINTER TO GET JOBNAME
     NOJOB      PRINTING EQ  4     SETSTATE ACTSTAT=ACTIVE.   CLEAR EXTDISP
     EXTRACT    PRINTING EQ  0     EXTRACT  EXTRACT JOBNAME FROM MESSAGE
     PARSEJOB   PRINTING EQ  0     PARSE    PARSE COMMA OFF END OF JOBNAME
     EXTDISP    PRINTING EQ  0     SETSTATE SET EXTENDED DISPLAY (PRINTING)
     STOP12                        STOP     DISPLAY PROC NOT TO SET STATE
    **END**




 F1=Help     F2=Split    F3=Exit     F4=Edit     F5=Find
 F7=Bkwd     F8=Forward  F9=Swap                F11=Right   F12=Min
```

The process steps are as follows:

Step 1. Use the SYSCMD macro to issue the $DU,&PRT command to display the information about the printer that is contained in the &PRT variable. You identify the printer by specifying the following parameter, PRT=*printer-resource-name*, when you specify the process on the Display and Heartbeat Details panel.

Define four $HASP603 message rules for the macro to check the status of the printer. Each rule checks for one of the following text strings and assigns the appropriate return codes:

- STATUS=ACTIVE (printing), return code 0
- STATUS=INACTIVE (not printing), return code 4
- STATUS=DRAINED (stopped), return code 5
- STATUS=DRAINING (stopping), return code 2

Figure 8-6 shows the completed panels.

Figure 8-6. Process Step—Checking Printer Status

```
SOLVPROD---- Automation Services : SYSCMD Macro Parameter Definition ----------
Command ===>                                                   Function=BROWSE

.- System Command -------------------------------------------------------------.
|                                                                               |
| Command ..... $DU,&PRT                                                        |
|                                                                               |
| Jobname .....                                                                 |
|                                                                               |
| Wait Time ... 10    Wait Time Expiry Return Code ... 8                        |
'-------------------------------------------------------------------------------'
.- Response Message Analysis --------------------------------------------------.
|                                                          S=Extended Filter    |
|           Message Text                                     Return Extended    |
|                                                            Code   Filter?     |
|   S__     $HASP603                                         0        YES       |
|   S__     $HASP603                                         4        YES       |
|   S__     $HASP603                                         5        YES       |
|   S__     $HASP603                                         2        YES       |
```

```
SOLVPROD------- Automation Services : Extended Message Filtering --------------
Command ===>                                                   Function=BROWSE

. Extended Filter Definition --------------------------------------------------.
|                                                                               |
| Message Text .......... $HASP603                                              |
| Jobname ..............          (Over-ride the default Jobname)              |
| Wildcard Character .... *                                                     |
'-------------------------------------------------------------------------------'
. Message Text Analysis ------------- Text for return code 0 ------------------.
|                                                                               |
|   Strt Word      Scan                                                         |
|   Pos  Num  Opr  Text                                                         |
| 1 001       CT   STATUS=ACTIVE                                                |
| 2                                                                             |
. Message Text Analysis ------------- Text for return code 4 ------------------.
|                                                                               |
|   Strt Word      Scan                                                         |
|   Pos  Num  Opr  Text                                                         |
| 1 001       CT   STATUS=INACTIVE                                              |
. Message Text Analysis ------------- Text for return code 5 ------------------.
|                                                                               |
|   Strt Word      Scan                                                         |
|   Pos  Num  Opr  Text                                                         |
| 1 001       CT   STATUS=DRAINED                                               |
. Message Text Analysis ------------- Text for return code 2 ------------------.
|                                                                               |
|   Strt Word      Scan                                                         |
|   Pos  Num  Opr  Text                                                         |
| 1 001       CT   STATUS=DRAINING                                              |
```

Step 2. Use the SETSTATE macro to set the actual state of the printer to STOPPING if the DISPPRTR step (Step 1) returns code 2 (see Figure 8-7).

Figure 8-7. Process Step—Setting Printer Actual State

```
SOLVPROD--- Automation Services : SETSTATE Macro Parameter Definition ---------
Command ===>                                                     Function=BROWSE

.- Resource Identification -----------------------------------------------------.
| Resource Name ..... &PRT                                                      |
| Resource Class ...+ PRT                                                       |
'-------------------------------------------------------------------------------'

.- Actual State ----------------------------------------------------------------.
| State ........... STOPPING (Active, Inactive, Failed, Degraded,             |
|                             Starting, Stopping or Unknown)                    |
| Description ...... &$RMMCMSGTEXT                                              |
'-------------------------------------------------------------------------------'

.- Extended Display ------------------------------------------------------------.
| Intensity ... LOW        (High or Low)                                        |
| Colour ...... GREEN      (Blue, Green, Pink, Red, Turquoise, White, Yellow) |
| Highlight ... NONE       (Blink, Reverse or None)                             |
|                                                                               |
| Text ........ ##RESET##                                                       |
'-------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

Step 3. Use the SETSTATE macro to set the actual state of the printer to INACTIVE if the DISPPRTR step (Step 1) returns code 5.

Step 4. Use the SETSTATE macro to set the actual state of the printer to ACTIVE if the DISPPRTR step (Step 1) returns code 4. The printer resource has an ACTIVE actual state but is not printing.

Step 5. If the DISPPRTR step (Step 1) returns code 0, you want to find out which job is being printed. Use the SYSCMD macro to issue the $DU,&PRT command again to find out about the job (see Figure 8-8). Check the returned $HASP603 message for the JOBNAME=*job-name* text string. You need this step because the STATUS keyword in Step 1 and the JOBNAME keyword appear in different lines of the $HASP603 message.

The step returns code 0 if the print job is identified or 4 if the print job has since completed.

Figure 8-8. Process Step—Checking Whether Printing Is in Progress or Completed

```
SOLVPROD---- Automation Services : SYSCMD Macro Parameter Definition ----------
Command ===>                                                   Function=BROWSE

.- System Command -------------------------------------------------------------.
|                                                                               |
| Command ..... $DU,&PRT                                                        |
|                                                                               |
| Jobname .....                                                                 |
|                                                                               |
| Wait Time ... 10    Wait Time Expiry Return Code ... 4                        |
'-------------------------------------------------------------------------------'
.- Response Message Analysis --------------------------------------------------.
|                                                          S=Extended Filter    |
|           Message Text                                     Return Extended    |
|                                                            Code   Filter?     |
|   ___     $HASP603                                         0        YES       |
|   ___                                                                         |
|   ___                                                                         |
|   ___                                                                         |
|   ___                                                                         |
|- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -|
' Message Text Analysis -------------------------------------------------------'
|                                                                               |
|   Strt Word      Scan                                                         |
|   Pos  Num  Opr  Text                                                         |
| 1 001       CT   JOBNAME=                                                     |
L_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _J
```

Step 6. If the PRINTING step (Step 5) returns code 4, set the actual state as for the ACTIVE step (Step 4). Then, jump to the STOP12 step (Step 10).

Step 7. If the PRINTING step (Step 5) returns code 0, use the EXTRACT macro to extract the JOBNAME=*job-name* information from the message (see Figure 8-9.

Figure 8-9. Process Step—Extracting the Name of the Job Being Printed

```
SOLVPROD--- Automation Services : EXTRACT Macro Parameter Definition ----------
Command ===>                                                     Function=BROWSE

.- Where to Extract From -------------------------------------------------------.
| Input .... &$RMMCMSGTEXT                                                       |
|           (Text String or Variable(s))                                         |
'--------------------------------------------------------------------------------'

.- Variable to hold Extracted String -------------------------------------------.
| Output ... TMPJOB       (Variable Name)                                        |
|                                                                                |
'--------------------------------------------------------------------------------'

.- How to do the Extraction ----------------------------------------------------.
| Starting Position ...     (Extraction to begin from this position. (1-256))    |
|                                                                                |
| Length to Extract ... 20  (Extract data for this length. (1-256))              |
|                                                                                |
| Find Text ........... JOBNAME=                                                 |
|       (Extraction to begin from the start of this text + Starting Position)    |
'--------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

Step 8. Use the PARSE macro to obtain *job-name* (see Figure 8-10).

Figure 8-10. Process Step—Parsing the Extracted Name

```
SOLVPROD---- Automation Services : PARSE Macro Parameter Definition -----------
Command ===>                                                     Function=BROWSE

.- String to be Segmented ------------------------------------------------------.
| Input .... &TMPJOB                                                             |
|            (Text String or Variable(s))                                        |
'--------------------------------------------------------------------------------'
.- Variables to be Created -----------------------------------------------------.
| Output ... *,JOBNAME                                                           |
|            (name or name1,name2... or name1(len),name2(len).. or prefix*)      |
'--------------------------------------------------------------------------------'
.- How to Segment the Input String ---------------------------------------------.
|    By Length ......          (YES or NO)                                       |
| OR By Delimiter ... =,       (Eight separate characters may be specified)      |
|                                                                                |
| Preserve Leading and Trailing Blanks? ..................... NO  (YES or NO)   |
|                                                                                |
| Create NULL Variables if consecutive delimiters found? .... YES (YES or NO)    |
|                                                                                |
| Remainder String Variable Name (If remainder required) ....                    |
'--------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

Step 9. Use the SETSTATE macro to set the actual state of the printer to ACTIVE and the ServiceView resource monitor status line to PRINTING JOBNAME=*job-name* (see Figure 8-11).

Figure 8-11. Process Step—Displaying the Status of a Printing Job

```
SOLVPROD--- Automation Services : SETSTATE Macro Parameter Definition ---------
Command ===>                                                   Function=BROWSE

.- Resource Identification -----------------------------------------------.
| Resource Name ..... &PRT                                                 |
| Resource Class ...+ PRT                                                  |
'--------------------------------------------------------------------------'

.- Actual State ----------------------------------------------------------.
| State ............ ACTIVE   (Active, Inactive, Failed, Degraded,        |
|                              Starting, Stopping or Unknown)              |
| Description ...... &$RMMCMSGTEXT                                         |
'--------------------------------------------------------------------------'

.- Extended Display ------------------------------------------------------.
| Intensity ... LOW        (High or Low)                                   |
| Colour ...... TURQUOISE  (Blue, Green, Pink, Red, Turquoise, White, Yellow) |
| Highlight ... NONE       (Blink, Reverse or None)                        |
|                                                                          |
| Text ........ PRINTING JOBNAME=&JOBNAME                                  |
'--------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

Step 10. Use the STOP macro to set the process return code to 12, telling ServiceView not to do anything to the actual state as the process has set the state already.

Note

If you do not require the job name, you can tell the region to set the actual state by passing the appropriate return code. The return codes are 0 for ACTIVE, 2 for STOPPING, and 5 for INACTIVE.

Testing a Process

Once you have defined a process, you can test it by executing it as a single task or by executing it in the interactive mode.

Testing a Process by Execution as a Single Task

Note

When you use the E action code to execute a process, the process is executed under the BSYS background user ID.

From the Process List panel, enter **E** beside a process to execute it as a single, independent task. The Optional Process Parameter Specification panel is displayed. Supply any parameters required by the process in the Parameters field, then press F6 (Action) to execute the process. When the process has executed, a processing log is displayed. This log contains the processing results.

Testing a Process Interactively

From the Process List panel, enter **I** beside a process to execute it in the interactive mode. The Process Definition panel for that process is displayed. You can:

- Enter **E** beside a step to execute only that step irrespective of the condition.
- Use F12 (Step) to execute a number of steps in sequence. Pressing F12 (Step) executes the next step in the sequence. The execution of each step depends on the condition specified for the step.
- Enter **L** beside an executed step to see the processing log. The log display is positioned at the latest entries relating to the selected step.
- Enter **P** beside a step to view the macro parameters.
- You can interactively edit and test the process steps as follows:

  a. Press F4 (Edit) to access the Interactive Edit function to edit the process steps.

  b. Modify the steps as required.

  c. When you complete the modifications, press F4 (OK) to return to the INTERACTIVE mode. (You can also press F3 (File) to return to that mode. Pressing F3 (File) saves the modifications.)

  d. Test the modified process.

  e. If the test is satisfactory, press F3 (Exit) and F3 (File) again to save the modified steps.

     If the test is not satisfactory, restart from Step a.

     You can press F12 (Cancel) to cancel the edit session.

Logging Process Activities

Process activities are written to the activity log while you are testing a process. You can, however, control the logging when a process is executed, for example, from a resource definition. Use the $LOG process parameter to control the logging as follows:

- BOTH—log activities in both full and summary form.
- FULL—log activities in full.
- NO (default)—do not log activities.
- SUMM—log activities in summary.

Maintaining Process Definitions

You can browse, update, copy, and delete process definitions from the Process List panel.

Backing Up Global Processes

To assist you with the maintenance of your global processes, you can create backup versions of your global process image. By creating a backup version of your global process image, you can perform the following:

- Update global process definitions in any version of a global process image
- Restore a global process definition from a backup global process image
- Merge two versions of a global process image

To create a backup version of a global process image, perform the following steps:

Step 1. Enter the **/ASADMIN.GPI** path to access the Global Process Image List (see Figure 8-12). If you have not created a backup before, there is only one global process image listed: $PROCESS 0001.

Note

The active global process image can only be $PROCESS 0001. $PROCESS 0001 cannot be deleted.

Step 2. Enter **C** beside the global process image you want to copy. The Global Process Image Definition panel is displayed (see Figure 8-13).

Figure 8-12. Copying a Global Process Image

```
SOLVPROD---------- ResourceView : Global Process Image List -------------------
Command ===>                                                   Scroll ===> PAGE

                           S/B=Browse U=Update C=Copy D=Delete L=List Processes
    Name      Ver    Short Description
 C  $PROCESS  0001   Base Global Process Image
```

Step 3. Enter a new Database Version, Short Description, and Long Description on the Global Process Image Definition Panel.

Figure 8-13. Creating a Backup Global Process Image

```
SOLVPROD-------- ResourceView : Global Process Image Definition ---------------
Command ===>                                                    Function=COPY


System Name ......... $PROCESS
Database Version .... 0002


Short Description ... Backup Global Process Image_
Long Description .... Global process image used for backups___________________
                      ______________________________________________________
                      ______________________________________________________
                      ______________________________________________________
```

Step 4. Press F3 (File) to save the backup version of the global process image. A copy in progress panel is displayed while the copy occurs. You are returned to the Global Process Image List with the backup version displayed in the list.

If the global process image you have specified already exists, the Confirm System Image Merge panel is displayed. See *Merging Two Global Process Images*, on page 8-21, for information on how to merge two global process images.

Updating Global Process Definitions in a Backup Global Process Image

You can access a list of all the global process definitions in any version of a global process image. From this list you can update any global process definition contained in the global process image.

To update a global process definition in the $PROCESS 0002 backup image created above, perform the following steps:

Step 1. Enter **L** (List Processes) beside the $PROCESS 0002 global process image in the Global Process Image List. The Global Process List panel is displayed showing all the global process definitions in that global process image.

Note

You can access the list of global processes for another version of the global process image by changing the version number on the Global Process List panel and pressing ENTER.

Step 2. Enter **U** beside the global process definition that you want to update. The Process Definition panel is displayed for that global process definition.

Step 3. Update the global process definition as required. See the information at the beginning of this chapter for details on how to update a global process definition.

Step 4. Press F3 (File) to save the changes. You are returned to the Global Process List panel.

Restoring a Global Process Definition From a Backup Global Process Image

If you have made changes to a global process definition and you are having trouble with its implementation, you can restore it from a previous version of the definition.

To restore global process definition $PROC01from $PROCESS 0002 to $PROCESS 0001, perform the following steps:

Step 1. Enter **C** beside $PROC01 in the global process list. The Process Definition panel is displayed (see Figure 8-14).

Figure 8-14. Restoring a Global Process Definition

```
SOLVPROD------- Automation Services : Process Definition -------Function=Update
Command ===>                                                     Scroll ===> PAGE

. Process Definition -------------------------------------------------------.
| System Name .+ $PROCESS  Version ...+ 0002                                |
| Name ......... $PROC01                                                    |
| Description .. updated desc                                               |
'---------------------------------------------------------------------------'
. Process Steps ------------------------------------------------------------.
|                                       D=Delete I=Insert P=Parms R=Repeat |
|                     Condition                                             |
|      StepName   Step/RC  Opr R/C   Macro    Description                   |
|      PARSEMSG                      PARSE    TOKENISE MESSAGE INTO PRT* VARS |
|      SETSTATE   PARSEMSG =   0     SETSTATE SET EXTENDED PRINT INFO       |
|     **END**                                                               |
|                                                                           |
|                                                                           |
|                                                                           |
|                                                                           |
|                                                                           |
|  F1=Help     F2=Split    F3=File     F4=Save                              |
|  F7=Backward F8=Forward  F9=Swap                F11=Right   F12=Cancel    |
|                                                                           |
'---------------------------------------------------------------------------'
```

Step 2. Change the Database Version from 0002 to 0001 and press F3 (File) to save the change. Because there is already a copy of the global process in the target global process image, the Confirm Copy Replace panel is displayed (see Figure 8-15).

Figure 8-15. Confirm Copy Replace Panel

```
SOLVPROD------- Automation Services : Confirm Copy Replace -----------------FTI


. Note ---------------------------------------------------------------------.
| This resource definition already exists in the target system image.        |
| If you continue this copy process with replace, the resource definition in  |
| the target system image will be replaced but any existing relationships are |
| preserved.                                                                 |
'----------------------------------------------------------------------------'

     Press ENTER to Confirm Copy Replace, Cancel Key to cancel copy process


            . Resource to Copy ---------------------------------------.
            |                    From              To                 |
            | System Name ...... $PROCESS          $PROCESS           |
            | System Version ... 0002              0001               |
            | Resource Class ... PRCESS            PRCESS             |
            | Resource Name .... $PROC01           $PROC01            |
            '-------------------------------------------------------'


 F1=Help      F2=Split
                          F9=Swap                                F12=Cancel
```

Step 3. Press ENTER to confirm the replace or F12 (Cancel) to cancel the request. You are returned to the Global Process List panel.

Changing a Global Process to a Local Process

You can change a global process to a local process while performing a copy on any global process in the global process selection list. To change global process PROC01 to a local process in the SYS01 system image, perform the following steps:

Step 1. Enter **C** beside PROC01 in the Global Process List panel. The Process Definition panel is displayed.

Step 2. Change the System Name to SYS01 and press F3 (File). You are returned to the global process list.

To view the new local process, access the list of processes for the system image that you copied it to.

Merging Two Global Process Images

You can merge two global process images and replace the active global process image with a backup version.

To merge global process images $PROCESS 0002 and $PROCESS 0001, complete the following steps:

Step 1. Enter **C** beside the $PROCESS 0002 on the Global Process Image List panel. The Global Process Image Definition panel is displayed.

Step 2. Change the Database Version number to 0001and press F3 (File). The Confirm System Image Merge panel is displayed (see Figure 8-16).

Figure 8-16. Confirm System Image Merge Panel for Global Process Images

```
SOLVPROD----- Automation Services : Confirm System Image Merge  ---------------


                            MERGE TWO SYSTEM IMAGES

     .-------------------------------------------------------------------.
     |  WARNING: THE SPECIFIED TARGET SYSTEM IMAGE ALREADY EXISTS.        |
     |           THIS REQUEST WILL MERGE ALL THE COMPONENTS FROM THE      |
     |           SOURCE SYSTEM IMAGE INTO THE TARGET SYSTEM IMAGE.        |
     |                                                                    |
     |              * Source System Image - $PROCESS-0002                 |
     |              * Target System Image - $PROCESS-0001                 |
     |                                                                    |
     |    Enter 'YES' to OVERLAY like-named components .... NO_           |
     |                                                                    |
     '-------------------------------------------------------------------'

     Press Confirm key to confirm request, Cancel key to cancel the request



 F1=Help      F2=Split                                          F6=Confirm
                           F9=Swap                             F12=Cancel
```

Step 3. Enter **YES** in the input field if you want to overlay like-named components.

Step 4. Press F6 (Confirm) to confirm the merge, or F12 (Cancel) to cancel the merge.

9

Defining and Maintaining Resource Templates

Products that use Automation Services include sample resource templates, which you can use to define commonly-used resources. You can modify the sample templates or create your own templates. You can create templates for the different resource types in each class of resource.

You can maintain several versions of templates as different $TEMPLAT system images. Each version can contain, besides the resource templates, the availability maps and processes used by the resource templates.

This chapter contains the following topics:

- Types of Resource Templates
- Setting Up Your Template System
- Defining and Maintaining Resource Templates
- Defining and Maintaining Availability Maps in a Template System Image
- Defining and Maintaining Processes in a Template System Image
- Converting a Resource Definition Into a Resource Template

Types of Resource Templates

There are the following types of resource templates:

- Product specific resource class templates
- USRCLS class template
- INTNL class template

The INTNL class templates are reserved for internal use.

Product Specific Resource Templates

Each product that uses Automation Services supplies its own templates for the supported resource classes. Refer to the user's guide for that product for information about the supported resource classes.

The templates provide the methods for operating the supported resources.

USRCLS Class Template

No sample USRCLS class templates are supplied. However, you can create your own templates to facilitate the definition of similar resources. The templates provide the methods for operating USRCLS class resources.

Setting Up Your Template System

Templates are defined in a $TEMPLAT system image. Your template system may contain different versions of templates. Group each version in a different $TEMPLAT system image.

Before you work on templates, copy the supplied templates to a different $TEMPLAT version. Start with version 0010, and leave versions 0001 through 0009 for future software updates.

Copy a $TEMPLAT system image as follows:

Step 1. Enter the **/RADMIN.T.I** path to access the list of $TEMPLAT system images. The System Image List panel is displayed.

Step 2. Enter **C** beside the system image you want to copy. The System Image Definition panel is displayed.

Step 3. Change the value in the Database Version field to uniquely identify the new copy (for example, 0010), and update the description fields as required.

Step 4. Press F3 (File) to start the copying process. The System Image Copy panel is displayed on your screen to advise you of the status of the copying process. When the copying process is complete, the System Image List panel is redisplayed.

Set up one $TEMPLAT system image version for general use. Review the templates to ensure that they are suitable for the resources on your system. The version to use is set in the OPSYSIDS parameter group under the NAMES category during region initialization. Enter the **/ICS** shortcut to access the Customization Parameters panel that enables you to access the parameter for update.

Merging $TEMPLAT System Images

Each product that uses Automation Services supplies its own templates for the supported resource classes. If you want to run different products in the same region, you need to merge the $TEMPLAT system images that contain those templates. See the section, *Merging System Images*, on page 4-6 for information about how to merge system images.

Defining and Maintaining Resource Templates

Caution

The supplied INTNL class resource templates are required for the region to function properly. Do *not* modify these templates.

After you have defined a system image, you can define resources in it. Predefined templates are supplied for supported resource types. The templates supply values for certain resource definition fields, and simplify the task of creating your own specific resource definitions. These templates contain generic information, including:

- Desired availability
- Activation commands and messages, or processes
- Inactivation and forced inactivation commands and messages, or processes
- Display commands and messages, or processes
- Unsolicited messages that indicate state changes

The information is similar to that specified in a resource definition.

If the supplied templates satisfy your requirements, you can use these templates to define commonly used resources. You can also modify the values supplied by a template or create new templates.

For example, you might want to create a template when you install a new version of CICS, and the new CICS resources use different commands or messages from the old version of CICS.

Accessing the Resource Template Definitions

Enter the **/RADMIN.R.T.R.*resource-class-name*** path to access the list of templates. For example, enter **/RADMIN.T.R.STC** to display the list of started task templates.

You can select a different version of the templates by changing the value in the Version field.

Defining a Resource Template

Note

The name of a template must contain alphanumeric, @, #, $, ., :, -, (, and) characters only. It must not be a number.

The panels used to add a resource template definition for a particular resource class are the same as the panels that you use when you add a resource definition for that class. You can define any information that will be used generically by a specific resource (for example, TSO or a silo tape unit). You use variables, strings of less than signs (<s), or strings of greater than signs (>s) to provide resource-specific details such as the resource name. You can also use the underline character (_) to define variables.

Using Variables

You can use a variable to supply the value for a field in the resource template definition.

Refer to Appendix D, *Variables*, for the list of variables and the information they supply.

Defining Variables

Variables in a template are substituted by their values when you apply the template to a resource definition. You can disable variable substitution—that is, you want the variable to appear in the resource definition, *not* the value of the variable. To disable variable substitution during application, replace the ampersand (&) in front of the variable name by the underline character (_). For example, if you specify _ZMSGTEXT in a template and apply the template to a resource definition, _ZMSGTEXT becomes &ZMSGTEXT in the resource definition.

Using Less Than Signs to Represent a Left-justified Fixed-length Variable Field Value

Some messages contain left-justified fixed-length fields for resource names. If the name is not of the maximum length, the name is left justified. To handle left-justified fixed-length fields, use less than signs (<s). Each < represents one character. You cannot use normal variables as they do not provide padding. For example, <<<<< represents a 5-character field with left justification.

Using Greater Than Signs to Represent a Right-justified Fixed-length Variable Field Value

Some messages contain right-justified fixed-length fields for resource names. If the name is not of the maximum length, the name is right justified. To handle right-justified fixed-length fields, use greater than signs (>s). Each > represents one character. You cannot use normal variables as they do not provide padding. For example, >>>>> represents a 5-character field with right justification.

Maintaining Resource Template Definitions

You can browse, update, copy, and delete resource template definitions. You can copy a resource template definition between or within $TEMPLAT system images.

Applying Updated Templates

You might have defined a number of resources by using a template, and that template has since been updated. You can use the AP action code to reapply the template to update those resource definitions. Proceed as follows:

Step 1. From the templates list, enter **AP** beside a template to display the Apply Template panel (Figure 9-1).

Figure 9-1. Applying a Template to Resources in a System Image

```
SOLVPROD----------- Automation Services : Apply Template ----------------------
Command ===>

.- Template Apply Parms -----------------------------------------------------.
|                                                                             |
|  Template Being Applied .... CICS                                           |
|  Merge, Override or Reset... MERGE___                                       |
|  Copy Availability Map? .... YES____  (Yes/No/Replace)                      |
|  Copy Processes? ........... YES____  (Yes/No/Replace)                      |
|                                                                             |
'-----------------------------------------------------------------------------'
```

Step 2. Specify how the updates are performed.

Step 3. Press F6 (Action) to list the system images that you can select to apply the template to.

Step 4. Identify the system images that contain the resource definitions to be updated, type **S** beside those images, and then press ENTER to apply the template to the included definitions.

Defining and Maintaining Availability Maps in a Template System Image

You can use the availability maps in a $TEMPLAT system image to define the desired availability in the resource templates belonging to that image. You can create new maps or change existing maps.

The procedures for creating and maintaining availability maps for resource templates are similar to the procedures for creating and maintaining maps for resource definitions. Refer to Chapter 7, *Defining and Maintaining Availability Maps*, for these procedures.

Accessing the Availability Map Definitions in a Template System Image

Access the availability map definitions in a $TEMPLAT system image from any panel as follows:

Step 1. Enter the **/RADMIN.T** path to access the Template Definition Menu.

Step 2. Type **A** at the Select Option ===> prompt and, if you want to use a different version of the $TEMPLAT system image, change the value in the Version field. Then, press ENTER.

The Availability Map List panel is displayed. The panel lists all the availability maps in the selected $TEMPLAT system image.

Defining an Availability Map in a Template System Image

The panel used to add an availability map for resource templates is the same as the panel that you use to add an availability map for resource definitions, and the procedure is similar.

Maintaining the Availability Map Definitions in a Template System Image

You can browse, update, copy, and delete the availability map definitions in a template system.

Defining and Maintaining Processes in a Template System Image

You can use the processes in a $TEMPLAT system image in a resource template belonging to the same image. You can create new processes or change existing processes.

The procedures for creating and maintaining processes for resource templates are similar to the procedures for creating and maintaining processes for resource definitions. Refer to Chapter 8, *Defining Processes to Implement Complex Operations*, for these procedures.

Accessing the Process Definitions in a Template System Image

Access the processes in a $TEMPLAT system image from any panel as follows:

Step 1. Enter the **/RADMIN.T** path to access the Template Definition Menu.

Step 2. Type **P** at the Select Option ===> prompt and, if you want to use a different version of the $TEMPLAT system image, change the value in the Version field. Then, press ENTER.

The Process List panel is displayed. The panel lists the processes in the selected $TEMPLAT system image.

Defining a Process in a Template System Image

The panel used to add a process for resource templates is the same as the panel that you use to add a process for resource definitions, and the procedure is similar.

Maintaining the Processes in a Template System Image

You can browse, update, copy, and delete the processes for resource templates. You can test and work with the processes interactively.

Converting a Resource Definition Into a Resource Template

You can convert a resource definition into a resource template to facilitate future definition of similar resources. After you are satisfied that a resource definition is working correctly, you can use the following procedure to convert the definition into a template:

Step 1. Use the Copy action to create another copy of the definition.

Step 2. Change the system name on the General Description panel to $TEMPLAT, and specify the version of the $TEMPLAT image into which you want to copy the definition in the Database Version field.

Step 3. Name the template on the General Description panel.

Step 4. Replace the resource names on the other definition panels by one of the following:

- &ZRMDBNAME if the name field is not of fixed length
- Less than signs (<s) if the name field is of fixed length with left justification—this typically occurs in the message text
- Greater than signs (>s) if the name field is of fixed length with right justification—this typically occurs in the message text

Note

Keeping the name length to less than the maximum number of characters enables you to easily recognize the fixed length name fields in a message. For example, a 7-character name is displayed with an extra space in an 8-character fixed length field.

Step 5. Replace the ampersand (&) in front of a variable by the underline character (_).

Step 6. File the definition. Any associated availability map and processes are also copied if they do not exist already in the specified $TEMPLAT system image.

10

Testing Definitions by Using Simulated Events

After you have defined and loaded a system image, you can use the event simulator to test its operations.

Note

The event simulator is part of the automation feature. Some products do not use the automation feature. These products use Automation Services to provide only knowledge base management and presentation services.

This chapter contains the following topics:

- What Is the Event Simulator?
- Generating Simulated Events
- Interpreting the Results of Event Simulation
- Maintaining Simulated Event Definitions

What Is the Event Simulator?

The event simulator enables you to correctly assess the impact of a loaded system image on the operations of the local system. The parameter group, EVENTSIM, controls the availability of the simulator.

By using the simulator, you can generate simulated events and review the returned results. A simulated event returns the expected results. It does not invoke the actual actions. The results of the simulation identify the following affected active definitions:

- Resource definitions
- EventView rules
- Consolidated console message profiles
- Other product-specific definitions and records (for example, SNA resource records of the NetMaster Automation product)

Generating Simulated Events

Generate simulated events from the Simulated Events List panel. To access the list, enter the **/EADMIN.E** path. You can then generate the simulated events in the following ways:

If the required event definition is …	**Then …**
Not on the list	Press F4 (Add) to define and generate the event
On the list	You can do one of the following: • Use the SV or SI action code to simulate one or more defined events. • Enter **ALL SI** at the Command ===> prompt to simulate all defined events.

Defining a Simulated Event

Define a simulated event from the Simulated Event List panel as follows:

Step 1. Enter the **/EADMIN.E** path to access the simulated event definitions.

Step 2. Press F4 (Add). You can also use the C action code to open a copy of an existing definition that you can modify.

Step 3. On the Simulate Message panel (Figure 10-1), specify the message you want to simulate and the type of information you want to be returned.

You can enter a question mark (?) in the Message Text field to display the list of messages learnt by the region. If you select a message from the list, the panel is automatically updated for any associated job name, route codes, and descriptor codes.

Figure 10-1. Defining a Simulated Event

```
SOLVPROD------------ Automation Services : Simulate Message --------------------
Command ===>                                                   Function=Add

. Message Attributes -------------------------------------------------------.
|                                                                            |
| Message Text .....+ $HASP170 PRT1     INTERRUPTED___________________________ |
|                     ________________________________________________________ |
| Job Name .......... JES2____                                               |
| Job Type .......... JOB  (Specify JOB, STC or TSU)                         |
| Message Type ...... WTO_  (Specify WTO or WTOR)                            |
| Route Codes ....... 7_______________________________________________________ |
| Descriptor Codes .. 4_______________________________________________________ |
|                                                                            |
| Note: If Jobname, Message Text or Jobtype change then results are cleared. |
'----------------------------------------------------------------------------'
. Simulation Control -------------------------------------------------------.
|                                                                            |
| EventView Rulesets ............. BOTH__  (HITS, MISSES, BOTH or NONE)     |
| Resources ...................... BOTH__    ''     ''     ''      ''       |
| Consolidated Console Profiles ... BOTH__    ''     ''     ''      ''       |
|                                                                            |
'----------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=File      F4=Save      F5=View      F6=Simulate
                           F9=Swap                                F12=Cancel
```

Step 4. After you have entered the information, proceed as follows:

If you …	Press …
Want to generate the simulated event	F6 (Simulate). If you want to save the results, you must press F3 (File) or F4 (Save).
Do not want to generate the simulated event now	F3 (File) to save the definition for later use.

Note

Filed message definitions are *not* retained across region restarts.

Interpreting the Results of Event Simulation

The results of event simulation are returned on the Simulation Results List panel (Figure 10-2).

Figure 10-2. Event Simulation Results

```
SOLVPROD------ Automation Services : Simulation Results List ------------------
Command ===>                                                   Scroll ===> CSR

                  S/B=Browse Definition U=Update Definition C=Collapse E=Expand
    Simulated Message Details:
     Message Text ... $HASP170 PRT1     INTERRUPTED
     Jobname ........ JES2           Jobtype ....... JOB  Message Type ... WTO
     Route Codes .... 7              Desc. Codes ... 4

    *************************** Simulation Results ****************************
    Dflt EventView Ruleset ..... $$$$$URS
         Default ruleset processing performed as per:
            Message Delivery ... YES           Perform Mods.? .. YES
            Perform Actions? ... YES           Log Activity? ... NO
            Collect Statistics?  YES           Learn New Msgs?   NO

    Miss No Consolidated Console profiles hit for the following reason:
            No Consolidated Console Profiles Hit

    Hit  PRT Resource Name ..... PRT1         JES Printer PRT1
         Monitor Message ....... $HASP170 PRT1*
         Extended Actions:
```

For the example in Figure 10-2, the results indicate that the:

- Message will be passed on by the $$$$$URS ruleset but no rules will be triggered
- PRT1 resource will become degraded but no actions will be invoked

If the results are not satisfactory for a displayed definition, you can use the U action code to update it. For example, if you enter U beside the PRT resource line, the Status Monitor Message Details panel is displayed. You can then update the appropriate resource message rule.

Summarizing the Results

When a simulated event affects many definitions, the results are displayed over several panels. You can summarize the results. Enter **ALL C** at the Command ===> prompt to present the results as a list of affected definitions (Figure 10-3).

Figure 10-3. Event Simulation Results Summary

```
SOLVPROD------ Automation Services : Simulation Results List ------------------
Command ===>                                                   Scroll ===> CSR

                  S/B=Browse Definition U=Update Definition C=Collapse E=Expand
    Simulated Message Details:
     Message Text ... $HASP170 PRT1     INTERRUPTED
     Jobname ........ JES2           Jobtype ....... JOB  Message Type ... WTO
     Route Codes .... 7              Desc. Codes ... 4

    ************************** Simulation Results ***************************
    Dflt EventView Ruleset ..... $$$$$URS
    Miss No Consolidated Console profiles hit for the following reason:
    Hit  PRT Resource Name ..... PRT1         JES Printer PRT1
    **END**
```

You can use the ALL E command to redisplay all details of all the results. You can use the C and E action codes to change the view of selected results.

Maintaining Simulated Event Definitions

You can browse, update, copy, and delete simulated event definitions. To delete all definitions, enter **ALL D** at the Command ===> prompt.

Note

If you update the message attributes, you are creating a new message. Previously stored simulation results are not retained.

11

Defining and Maintaining Filters for the Status Monitor

The status monitor displays the status of defined services and the status of defined resources in the currently active system images. The display is in the form of a list. A status monitor can be customized to display only the services and resources of interest.

You customize a status monitor by using status monitor filters. You can selectively view different groups of services and resources by swapping filters.

This chapter contains the following topics:

- Implementing the Status Monitor Filters
- Accessing Status Monitor Filter Definitions
- Defining a Status Monitor Filter
- Maintaining Status Monitor Filter Definitions

Implementing the Status Monitor Filters

After you define a filter, that filter can be used to customize a status monitor screen. For example, you can define a filter that, when used, causes the status monitor to display only those services and resources that are applicable to the accounts payable business function.

A status monitor filter uses a Boolean expression, which you define on the Status Monitor Filter panel, to determine what to display on the monitor. You restrict the display by using the service and resource attributes such as names and status.

When you save a filter definition in the knowledge base, the definition propagates automatically to all the connected regions—that is, the definition is global.

Accessing Status Monitor Filter Definitions

Enter the **/ASADMIN.F** path to access the list of status monitor filter definitions. The Status Monitor Filter List panel is displayed. Figure 11-1 is an example.

Figure 11-1. Status Monitor Filter List Panel

```
SOLVPROD----- Automation Services : Status Monitor Filter List ----------------
Command ===>                                                     Scroll ===> 10

                                              S/B=Browse U=Update C=Copy D=Delete
    Filter Name   Description
    DASD          DASD devices
    INIT          Initiators
    INTNL         Internal resources
    JOB           Batch jobs
    LINE          JES2 lines
    NONE          Null filter
    NOTOK         Resources that are not O.K
    PRT           Printers
    RESOURCEVIEW  All Resources
    SERVICEVIEW   Monitor Services Only
    STC           Started tasks
```

The panel displays the list of status monitor filter definitions in the knowledge base. You can add a new definition, or browse, update, copy or delete an existing definition.

Defining a Status Monitor Filter

Add a status monitor filter definition as follows:

Step 1. Access the Status Monitor Filter List panel.

Step 2. Press F4 (Add) to add a filter definition. The Status Monitor Filter panel is displayed.

Note

If you change your mind and do *not* want to add the filter, press F12 (Cancel) to cancel the operation any time before Step 5.

Step 3. Complete the Name and Description fields in the Filter Definition window to identify the new filter. See the online help for a description of the fields.

Step 4. Specify a Boolean expression in the Filter Expression window to define the filter. See the section, *Defining the Status Monitor Filter Expression*, on page 11-5 for information about how to construct the Boolean expression.

Step 5. Press F3 (File) to file the new definition when you finish defining the filter.

Using the Status Monitor Filter Panel

The Status Monitor Filter panel specifies the details of a status monitor filter. Figure 11-2 is an example of the panel. The operation you are performing is displayed at the top right-hand corner of the panel (for example, Function=UPDATE).

Figure 11-2. Status Monitor Filter Panel

```
SOLVPROD------ Automation Services : Status Monitor Filter -----Function=UPDATE
Command ===>                                                    Scroll ===> 10

+ Filter Definition -------------------------------------------------------+
| Name ......... ATTENTION                                                 |
| Views ........ ________ ________ ________ ________  ________  ________   |
| Description .. RESOURCES THAT ARE IN ATTENTION MODE                      |
| Last Updated at 22.09.04 on WED 21-MAY-1997 by USER01                    |
+--------------------------------------------------------------------------+
+ Filter Expression --------------------------------------------------------+
|                                                                          |
|                                                D=Delete I=Insert R=Repeat |
|      "(" Field   Opr Value                                Gen ")" Bool   |
|       ___ LOGSTAT =   "ATTENTION"                         ___ ___ ___    |
|     **END**                                                              |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|  F1=Help     F2=Split    F3=File     F4=Save                             |
|  F7=Bkwd     F8=Forward  F9=Swap                            F12=Cancel   |
+--------------------------------------------------------------------------
```

The panel displays two windows. The Filter Definition window identifies the filter by name and description, and the Filter Expression window specifies the Boolean expression that defines the filter.

Status Monitor Views

A view customizes your status monitor for the specific purpose of monitoring certain classes of objects. Each view has associated with it a selected set of filters and display formats.

To see the supported views, enter **?** in a View field.

Defining the Status Monitor Filter Expression

Use the Filter Expression window on the Status Monitor Filter panel (Figure 11-2) to specify the Boolean expression that defines the filter. The expression uses service and resource attributes as criteria to determine what to display on the status monitor. The section, *Relational Operators*, on page G-2 describes the relational operators that you can use to define the expression.

To display the list of valid values for a field, enter a question mark (?) in the field.

Use the D (Delete), I (Insert), and R (Repeat) action codes to help you enter the expression:

Delete	Deletes the selected line.
Insert	Inserts a blank line after the selected line.
Repeat	Repeats a selected line.

Example

In this example, you define a filter called SERVICEALERT that enables an operator to monitor services that have a DEGRADED, FAILED, or UNKNOWN logical state. Figure 11-3 shows the completed Status Monitor Filter panel.

Figure 11-3. Example—Service Filter Definition for the Status Monitor

```
SOLVPROD------- Automation Services : Status Monitor Filter ----Function=BROWSE
Command ===>                                                   Scroll ===> CSR

. Filter Definition -------------------------------------------------------.
| Name ......... SERVICEALERT                                               |
| Views ........ ________ ________ ________ ________ ________  ________    |
| Description .. Services in DEGRADED, FAILED, or UNKNOWN state             |
| Last Updated at 15.09.30 on WED 28-MAY-1997 by USER01                     |
'---------------------------------------------------------------------------'
. Filter Expression -------------------------------------------------------.
|                                                                           |
|                                                                           |
|      "(" Field   Opr Value                                   Gen ")" Bool |
|          CLSNAME =   "SVC"                                           AND  |
|      (   LOGSTAT =   "DEGRADED"                                      OR   |
|          LOGSTAT =   "FAILED"                                        OR   |
|          LOGSTAT =   "UNKNOWN"                                   )        |
|     **END**                                                               |
|                                                                           |
|                                                                           |
|                                                                           |
|  F1=Help      F2=Split     F3=Exit      F4=Edit      F5=Find      F6=Refres |
|  F7=Backward  F8=Forward   F9=Swap                               F12=Max  |
'---------------------------------------------------------------------------
```

The filter expression causes a status monitor to display only services that have the DEGRADED, FAILED, or UNKNOWN logical state.

Maintaining Status Monitor Filter Definitions

You can browse, update, copy, and delete filter definitions from the Status Monitor Filter List panel.

If the Filter Expression window does not fully display the Boolean expression while you are browsing a filter definition, press F12 (Max) to expand the window.

Note

After you update a filter definition, an operator who is already using that filter does not see the update. In order to use the updated filter, the operator needs to enter the REFILTER command.

12

Defining and Maintaining Resource Groups, Icons, and Icon Panels for the Graphical Monitor

The graphical monitor presents the status of services and resources in icons on an icon panel. An icon can represent a group of services, a group of resources, or a group of both services and resources. A graphical monitor can be customized to display the icons of interest.

Note

The graphical monitor is not available to products that do not use the automation feature.

You customize a graphical monitor by using icon panels. You can change the icon panel to obtain a different view of the monitored systems networks. By zooming in on an icon, you can selectively view the required group of services and resources.

This chapter contains the following topics:

- Customizing the Graphical Monitor
- Defining and Maintaining Resource Groups for Icons
- Defining and Maintaining Icons
- Defining and Maintaining Icon Panels
- Editing a Generated Icon Panel

Customizing the Graphical Monitor

You define resource groups, icons, and icon panels to enable customization of the graphical monitor. You arrange icons on icon panels and attach resource groups to the icons so that each icon on the panel represents a group of services and resources. After you generate an icon panel, an operator can use that panel to customize the graphical monitor.

You generate an icon panel as follows:

Step 1. Define the required resource groups and icons.

Step 2. Define the icon panel by positioning the icons on the panel and attaching resource groups to the icons.

Step 3. Generate the icon panel description file. A panel description file contains statements that define the display format of the icon panel.

When you save a resource group, icon, or icon panel definition in the knowledge base, or generate an icon panel description file, the definition or the description file propagates automatically to all the connected regions. That is, the definition or the generated icon panel is global.

Defining and Maintaining Resource Groups for Icons

A resource group represents a group of services and resources that you have defined in the knowledge base. To define a resource group, use one of the following methods:

- Specify an icon panel. The panel displays icons representing other resource groups. Use the Zoom Icon Panel Definition panel to specify the icon panel.
- Specify a group of up to 16 services and resources. You identify the services and resources by class and name. Thus, the identified resources are independent of system images. In a multisystem environment, the specified class and name points to all resources of the class and name in all the system images that are loaded in the linked regions. You can, however, specifically exclude remote resources. Use the Resource Filter Definition panel to specify the services and resources to be grouped.
- Specify a resource group filter. A resource group filter uses a Boolean expression to define a group of services and resources. You group the services and resources by their static attributes such as names and parent system images. Use the Resource Group Filter Definition panel to define the Boolean expression.

Accessing Resource Group Definitions

enter the **/GADMIN.G** path to access the list of resource group definitions. The Resource Groups List panel is displayed. Figure 12-1 is an example.

Figure 12-1. Resource Groups List Panel

```
SOLVPROD--------- Automation Services : Resource Groups List ------------------
Command ===>                                                   Scroll ===> 10

                                              S/B=Browse U=Update C=Copy D=Delete
    Name          Description
    BACK          BACKUP
    CICS          CICS
    CMNM          CMNM regions
    CNM           NETVIEWS
    CSNDM         CONNECT:Directs
    CSNM          CSNM regions
    DASD          DASD devices
    DCE           DCE tasks
    DENM          DENM regions
    DSN           DB2 regions
    FNMS          NPM regions
    INIT          Initiators
    JES           JES subsystem
    JOB           Batch jobs
    LINE          JES lines
    PERP          Peripherals
    PRT           Printers
```

The panel displays the list of resource group definitions in the knowledge base. You can add a new definition, or browse, update, copy or delete an existing definition.

Defining a Resource Group

Add a resource group definition as follows:

Step 1. Access the Resource Groups List panel.

Step 2. Press F4 (Add) to add a group definition. The Resource Group Definition panel is displayed.

Note

If you change your mind and do *not* want to add the group, press F12 (Cancel) to cancel the operation any time before Step 6.

Step 3. Complete the Name and Description fields to identify the new group.

Step 4. Select one of the following options to define the group:

> **Note**
> Options B and C are related. You can use option B to specify the services and resources in the group directly. If you then select option C, the specification defined by using option B is expanded into a Boolean expression.

- Select option A to specify an icon panel. The Zoom Icon Panel Definition panel is displayed. Proceed to Step a.
- Select option B to specify a group of resources by class and name. The Resource Filter Definition panel is displayed. Proceed to Step b.
- Select option C to specify a resource group filter. The Resource Group Filter Definition panel is displayed. Proceed to Step c.

Step 5. Depending on the option you select, proceed as follows:

a. Specify the name of a generated icon panel in the Zoom Icon Panel Name field. You can enter a question mark (?) in the field to access the icon panel prompt list from which you can select the required panel.

   After you specify the name, proceed to Step 6.

b. Identify the resources by class and name in the ClassDsc and RscName fields. You can enter a question mark (?) in the fields to access the resource class and resource name prompt lists from which you can select the required class and name.

   If you want to exclude the resources from remote systems, specify **Y** (yes) in the Exclude Remote System Resource field. The default is NO.

   After you identify the resources, proceed to Step 6.

c. Press F10 (EditFltr) to edit the filter. See the online help for a description of the fields.

   Specify the Boolean expression in the Filter Expression window to define the filter. See the section, *Defining the Resource Group Filter Expression*, on page 12-5 for information about how to construct the Boolean expression.

   Press F3 (OK) to exit the edit mode, then proceed to Step 6.

Step 6. Press F3 (File) to file the new definition when you finish defining the group.

Using the Resource Group Filter Definition Panel

The Resource Group Filter Definition panel specifies the details of a resource group. Figure 12-2 is an example of the panel. The operation you are performing is displayed at the top right-hand corner of the panel (for example, Function=UPDATE).

Figure 12-2. Resource Group Filter Definition Panel

```
SOLVPROD--------Automation Services : Resource Group Filter --------Func=UPDATE
Command ===>                                                   Scroll ===> 10

+ Filter Definition -------------------------------------------------------+
| Name ......... $ICABCD                                                   |
| Description .. RESOURCE GROUP "ABCD" DIRECT FILTERING                    |
| Last Updated at 09.58.43 on FRI 23-MAY-1997 by USER01                    |
+--------------------------------------------------------------------------+
+ Filter Expression --------------------------------------------------------+
|                                                                          |
|                                                D=Delete I=Insert R=Repeat |
|      "(" Field   Opr Value                                Gen ")" Bool   |
|      (   CLSNAME EQ  "STC"                                 ___ ___ AND   |
|          NAME    NE  "PCICS1"                              ___ )   ___   |
|     **END**                                                              |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|  F1=Help     F2=Split    F3=OK                                           |
|  F7=Bkwd     F8=Forward  F9=Swap                           F12=Cancel   |
+--------------------------------------------------------------------------+
```

The panel displays two windows. The Filter Definition window identifies the filter, and the Filter Expression window specifies the Boolean expression of the filter.

Defining the Resource Group Filter Expression

Use the Filter Expression window on the Resource Group Filter Definition panel (Figure 12-2) to specify the Boolean expression that defines the filter. The expression uses service and resource attributes to determine what belongs to the group. The section, *Relational Operators*, on page G-2 describes the relational operators that you can use to define the expression.

Use the D (Delete), I (Insert), and R (Repeat) action codes to help you enter the expression:

Delete	Deletes the selected line.
Insert	Inserts a blank line after the selected line.
Repeat	Repeats a selected line.

Maintaining Resource Group Definitions

You can browse, update, copy, and delete group definitions from the Resource Group List panel.

Note

During an update, if the resources in the resource group are specified by using option C, you have no access to option B.

Except as noted above, you can change the method of definition during an update. Saving a definition by a new method automatically overrides the definition by the current method.

Defining and Maintaining Icons

An icon is a graphic that you can use to represent resource groups on the graphical monitor. You use icons to build the icon panels. You position one or more icons on a panel and attach resource groups to the icons, one group for each icon. When used, an icon displays a status determined by the status of the underlying group members. An operator can zoom in from an icon on a graphical monitor to the underlying icons until finally a group of services and resources is displayed, as determined by the attached resource groups. Use the Icon Editor to define an icon.

Accessing the Icon Definitions

Enter the **/GADMIN.I** path to access the list of icon definitions. The Icon List panel is displayed. Figure 12-3 is an example.

Figure 12-3. Icon List Panel

```
SOLVPROD------------- Automation Services : Icon List ---------------$ICON-0001
Command ===>                                                   Scroll ===> 10

                                            B/S=Browse U=Update C=Copy D=Delete
    Name         Description
    ICON1        General resource group ICON
    ICON2        General resource group ICON
    RMDFLT       Icon used on $RMDEFAULT panel
    **END**
```

The panel displays the list of icon definitions in the knowledge base. You can add a new definition, or browse, update, copy or delete an existing definition. You can also edit a definition from the Icon Panel Generator panel (see the section, *Editing an Icon on the Icon Panel*, on page 12-20).

Defining an Icon

Add an icon definition as follows:

Step 1. Access the Icon List panel.

Step 2. Press F4 (Add) to add an icon definition. The Icon Editor panel is displayed.

Note

If you change your mind and do *not* want to add the icon, press F12 (Cancel) to cancel the operation any time before Step 5.

Step 3. Complete the Name and Description fields to identify the new icon, and specify the size of the icon in the Icon Height and Icon Width fields. If you change the default size, press ENTER to update the shape of the icon in the Edit Area window. See the online help for a description of the fields.

Step 4. Specify values you want to display on the icon. See the section, *Editing the Icon*, on page 12-9 for a description of what you can do in the Edit Area window.

Step 5. Press F3 (File) to file the new definition when you finish defining the icon.

The Icon Editor panel specifies the details of an icon. Figure 12-4 is an example of the panel. The operation you are performing is displayed at the top right-hand corner of the panel (for example, Function=UPDATE).

Figure 12-4. Icon Editor Panel

```
SOLVPROD-------------- Automation Services : Icon Editor ----------------------
Command ===>                                                   Function=UPDATE

  Name .... CVNEW       Description .... A new icon

+ Edit Area ----------- Icon Height 7   Icon Width 20 --+ ACT Actual State
|                                                       | CLD Class Name
|                                                       | CMD Input Field
|                                                       | CNT Resource Counts
|                                                       | DES Desired State
|                  ___  RESOURCE NAME                   | DSC Description
|                  ___ ====================            | KWD User Keyword
|                  ___  DESIRED STATE                   | LGS Logical State
|                  ___ ====================            | MOD AutomationMde
|                  ___   INPUT FIELD                    | NME Resource Name
|                  ___ ====================            | PAD Blank to Clear
|                                                       | SYS System Name
|                                                       | TOT ResourceTotal
|                                                       | TXT Free Form Text
|                                                       | VER SystemVersion
+-------------------------------------------------------+

 F1=Help      F2=Split     F3=File      F4=Save      F5=Clear
                           F9=Swap                                F12=Cancel
```

The panel specifies the following information:

- Name and description of the icon
- Size of the icon (height and width)
- Actual icon representation

The Edit Area window specifies the values you want to display on the icon. See the next section, *Editing the Icon*, for a description of what you can do in the Edit Area window.

Editing the Icon

Use the Edit Area window on the Icon Editor panel (see Figure 12-4 on page 12-8) to specify what you want displayed on the icon.

The icon contains the number of lines specified in the Icon Height field. Use the 3-character codes listed to the right of the Edit Area window to specify the values you want displayed on the icon. To use a code, enter the code in a line field. You can use the code on any line, irrespective of whether the line is blank or not. Except for the TXT code, executing a code on a line overrides what is already there.

You can type codes in more than one line field, then press ENTER to execute the codes.

Pressing F5 (Clear) clears the icon. Use the PAD code to clear a line.

Codes

ACT

Puts the actual state of the group member that is in the worst logical state on the icon when displayed on the graphical monitor. The code expands to Actual State when executed.

Use the TXT code to enter a description to identify the information. For example, without descriptive text, the icon might display DEGRADED; with descriptive text, the icon can display Actual State:DEGRADED, where Actual State is entered by using the TXT code.

CLD

Puts the class of the group member that is in the worst logical state on the icon when displayed on the graphical monitor. The code expands to Class Name when executed.

Use the TXT code to enter a description to identify the information.

CMD

Puts a command entry field on the icon when displayed on the graphical monitor. The code expands to Input Field when executed. An icon can have only one command entry field.

CNT

Puts the total number of group members that are in a particular logical state on the icon. When you execute CNT, a field is displayed on your icon to enable you to enter one of the following keywords to indicate the logical state:

Keyword	For Logical State
#ATT	ATTENTION
#DEG	DEGRADED
#FAI	FAILED
#INE	INERROR
#OK	OK
#PEN	PENDING
#STA	STARTING
#STO	STOPPING
#UNK	UNKNOWN

The code expands to # *logical-state* (for example, # ATTENTION).

Use the TXT code to enter a description to identify the information you want to display.

DES

Puts the desired state of the group member that is in the worst logical state on the icon when displayed on the graphical monitor. The code expands to Desired State when executed.

Use the TXT code to enter a description to identify the information.

DSC

Puts the description of the group member that is in the worst logical state on the icon when displayed on the graphical monitor. The code expands to Description when executed.

Use the TXT code to enter a description to identify the information.

KWD

Enables you to specify one of the user-defined variables, defined by keywords for the monitored messages in the resource definitions. You can also specify the EXTD keyword for the defined extended display. See the section, *Defining Extended Display Attributes*, on page 5-39 for information about user-defined variables and extended display.

If the variable or the extended display is defined for the resource that is in the worst logical state within the attached resource group, the value of the variable or the extended display appears on the icon when displayed on the graphical monitor.

The code expands to UserKWD(*kkkk*), where *kkkk* is the keyword name or EXTD.

Use the TXT code to enter a description to identify the information you want to display.

LGS

Puts the logical state of the group member that is in the worst logical state on the icon when displayed on the graphical monitor. The code expands to Logical State when executed.

Use the TXT code to enter a description to identify the information.

MOD

Puts the operation mode of the group member that is in the worst logical state on the icon when displayed on the graphical monitor. The code expands to AutomationMde when executed.

Use the TXT code to enter a description to identify the information.

NME

Puts the name of the group member that is in the worst logical state on the icon when displayed on the graphical monitor. The code expands to Resource Name when executed.

Use the TXT code to enter a description to identify the information.

PAD

Enables you to fill a line with a repeating character. When you execute PAD, the PadChar=> prompt is displayed. Supply the required character, and press ENTER. The character fills the line. You can fill the line with blanks if you leave the prompt field blank and press ENTER.

SYS

Puts the name of the system image containing the resource that is in the worst logical state within the attached resource group on the icon when displayed on the graphical monitor. If a service has the worst logical state, SYS is $SERVICE. The code expands to System Name when executed.

Use the TXT code to enter a description to identify the information.

TOT

Puts the total number of group members on the icon when displayed on the graphical monitor. If the resource group attached to the icon points to an icon panel, the parameter gives the total number of resource groups on that icon panel. The code expands to ResourceTotal when executed.

Use the TXT code to enter a description to identify the information.

TXT

Enables you to enter text in a line (for example, to identify the attached resource group). The entered text is suffixed with a colon (:).

If you execute TXT on a line that contains the expanded form of another code (for example, Actual State), a field is displayed on the line to enable you to add text in front of what is already there. The exception is when the line displays Input Field. Executing TXT on a line that displays Input Field overrides the whole line for text entry.

VER

Puts the version of the system image containing the resource that is in the worst logical state within the attached resource group on the icon when displayed on the graphical monitor. The code expands to SystemVersion when executed.

Use the TXT code to enter a description to identify the information.

Example

In this example, an icon, EFTPOS, is defined for the group of services and resources that support electronic funds transfer. The finished icon as it appears to an operator is shown in Figure 12-5:

Figure 12-5. Displayed Icon

```
ELECTRONIC FUNDS TRANSFER
Actual State:DEGRADED
Desired State:ACTIVE

Operation Mode:AUTOMATED

Worst State Member
System:$SERVICE
Name:CREDITAUTH
```

To define the icon, proceed as follows:

Step 1. Enter the **/GADMIN.I** path to access the Icon List panel.

Step 2. Press F4 (Add) to add an icon definition. The Icon Editor panel is displayed.

Step 3. Enter **EFTPOS** in the Name field and a description in the Description field (for example, Electronic funds transfer).

Step 4. You want the icon size to be 10 lines by 30 characters. Change the icon width to 30, and press ENTER to update the shape of the icon.

Step 5. Enter **TXT** in the first line field. A text field is displayed in the icon.

Step 6. Enter **ELECTRONIC FUNDS TRANSFER** in the text field.

Step 7. Enter **CMD** in the second line field to create the command entry field.

Step 8. Enter **ACT** in the third line field.

Step 9. To add the description to the field, enter **TXT** in the line field. Enter **Actual State** in the displayed text field. A colon is inserted automatically after the entered text.

Step 10. Repeat steps 8 and 9, using the appropriate codes, for the desired state (line 4), the operation mode (line 6), and the identity of the group member that has the worst logical state (lines 9 and 10).

Use steps 5 and 6 to enter the Worst State Member label (line 8).

Figure 12-6 shows the completed Icon Editor panel.

Figure 12-6. Example—Icon Definition

```
SOLVPROD-------------- Automation Services : Icon Editor ----------------------
Command ===>                                                   Function=ADD

  Name .... EFTPOS     Description .... Electronic funds transfer

. Edit Area ----------- Icon Height 10   Icon Width 30 --. ACT Actual State
|                                                         | CLD Class Name
|                                                         | CMD Input Field
|          ___     ELECTRONIC FUNDS TRANSFER              | CNT Resource Counts
|          ___           Input Field                      | DES Desired State
|          ___    Actual State:Actual State               | DSC Description
|          ___   Desired State:Desired State              | KWD User Keyword
|          ___                                            | LGS Logical State
|          ___   Operation Mode:AutomationMde             | MOD AutomationMde
|          ___                                            | NME Resource Name
|          ___       Worst State Member                   | PAD Blank to Clear
|          ___       System:System Name                   | SYS System Name
|          ___       Name:Resource Name                   | TOT ResourceTotal
|                                                         | TXT Free Form Text
|                                                         | VER SystemVersion
'---------------------------------------------------------'

 F1=Help      F2=Split     F3=File      F4=Save      F5=Clear
                           F9=Swap                              F12=Cancel
```

Step 11. Press F3 (File) to file the definition when you finish with the panel.

Maintaining Icon Definitions

You can browse, update, copy, and delete icon definitions from the Icon List panel.

Defining and Maintaining Icon Panels

An icon panel defines what is displayed on the graphical monitor. You arrange icons on the panel and attach resource groups to the icons.

When you create an icon panel, you create two things: an icon panel definition and the icon panel description file. An operator uses the panel to customize the graphical monitor. You can generate an icon panel (that is, the description file) only if all the icons on the corresponding icon panel definition have attached resource groups. Use the Icon Panel Generator to define and generate the icon panel.

Caution

Icon panels defined on a 3270 model 4 or equivalent terminal cannot be used on model 3 and model 2 terminals; icon panels defined on a model 3 terminal cannot be used on model 2 terminals.

Accessing Icon Panel Definitions

Enter the **/GADMIN.P** path to access the list of icon panel definitions. The Icon Panel Definition List panel is displayed. Figure 12-7 is an example.

Figure 12-7. Icon Panel Definition List Panel

```
SOLVPROD------ Automation Services : Icon Panel Definition List -----$ICON-0001
Command ===>                                                    Scroll ===> 10

                                              B/S=Browse U=Update C=Copy D=Delete
    Name         Description
    $RMDEFAULT   Default icon monitor
    ACCOUNT      Accountants services
    ACCOUNTSPAY  Accounts payable service
    ACCOUNTSREC  Accounts receive service
    ASDF         ASDF
    CCVSTMSOLV13 The SOLV13 icon
    CCVSTMSOLV14 The SOLV14 icon
    DPTEST       Critical services (DP)
    QAIP2        qrm22011 resources
    **END**
```

The panel displays the list of icon panel definitions in the knowledge base. You can add a new definition, or browse, update, copy, or delete an existing definition.

Defining and Generating an Icon Panel

Add an icon panel definition as follows:

Step 1. Access the Icon Panel Definition List panel.

Step 2. Press F4 (Add) to add an icon panel definition. The Icon Panel Generator Initial Help panel is displayed.

Note

Pressing F4 (Remove Help Screen) exits and removes permanently the help panel. That is, the help panel does not appear the next time you work on an icon panel definition.

Step 3. When you finish reading the help text, press ENTER to exit. The Icon Panel Generator panel is displayed.

Note

If you change your mind and do *not* want to add the panel, press F12 (Cancel) to cancel the operation any time before Step 6.

Step 4. Complete the Name and Desc fields to identify the new panel. See the online help for a description of the fields.

Step 5. Use the function keys to create your panel. The left limit of the icon placement area is column 2, and the top limit of the icon placement area is row 5. The right and bottom limits are dependent on the size of your screen and the width and height of the icon.

Step 6. Press F3 (File) to generate the new panel and file the new definition when you finish defining the panel.

If an icon in the panel definition does not have an attached resource group, you cannot generate the new panel. A message is displayed on your screen to advise you of the fact. You can either attach any missing resource groups so that you can generate the panel or press F3 (File) again to file the definition without generating the panel.

Using the Icon Panel Generator Panel

The Icon Panel Generator panel specifies the details of an icon panel. Figure 12-8 is an example of the panel. The operation you are performing is displayed at the top right-hand corner of the panel (for example, Function=UPDATE).

Figure 12-8. Icon Panel Generator Panel

```
SOLVPROD------- Automation Services : Icon Panel Generator --------------------
Command ===>                                                     Function=UPDATE

           Name ... ACCOUNTSPAY   Desc ... Accounts payable system


    DESIRED STATE                 DESIRED STATE
 ====================          ====================
    RESOURCE NAME                 RESOURCE NAME
 ====================          ====================
   ******************            ******************
 ====================          ====================




 F1=Help      F2=Split     F3=File      F4=Save      F5=CutIcon   F6=PutIcon
 F7=PickIcon  F8=EditIcon  F9=Swap     F10=Query    F11=PickGrp  F12=Cancel
```

The panel specifies the following information:

- Name and description of the icon panel
- Actual icon panel representation

The area from column 2 to the right and from row 5 downward contains the icons you want to display on the graphical monitor.

Use the function keys on the Icon Panel Generator panel (Figure 12-8) to specify what you want to display on the graphical monitor.

Adding an Icon to the Icon Panel

Add an icon to the icon panel as follows:

Step 1. Move the cursor to fix the position of the top left corner of your icon. You must place the cursor in an area not already occupied by another icon.

Step 2. Press F7 (PickIcon) to display the list of defined icons. The Icon List panel is displayed.

Note

While you are on the Icon List panel, you can create and maintain icons as described in the section, *Defining an Icon*, on page 12-7.

Step 3. Enter **S** beside the icon you want to add to the icon panel. The Icon Panel Generator panel is redisplayed. The selected icon is positioned with its top left corner at the cursor.

Note

After you pick an icon, you can move the cursor to another position and press F6 (PutIcon) to duplicate the icon on the icon panel. You can thus quickly position multiple icons with the same attributes on the panel.

Step 4. Press F11 (PickGrp) to attach a resource group to the icon. The Resource Groups List panel is displayed.

Note

While you are on the Resource Groups List panel, you can create and maintain resource groups as described in the sections following the section, *Defining and Maintaining Resource Groups for Icons*, on page 12-2.

Step 5. Enter **S** beside the group you want to attach to the icon. The Icon Panel Generator panel is redisplayed.

You have added an icon with an attached resource group to the icon panel.

Attaching a Resource Group to an Icon on the Icon Panel

You can attach resource groups to icons on the Icon Panel Generator panel. You can change a resource group attachment by attaching another group to the icon. Attach a resource group to an icon on the icon panel as follows:

Step 1. Move the cursor to within the icon to which you want to attach a resource group.

Step 2. Press F11 (PickGrp) to attach a resource group to the icon. The Resource Groups List panel is displayed.

Step 3. Enter **S** beside the group you want to attach to the icon. The Icon Panel Generator panel is redisplayed.

Duplicating an Icon on the Icon Panel

Note

Duplicating an icon on the icon panel copies only the icon, not the attached resource group.

Duplicate an icon on the icon panel as follows:

Step 1. Move the cursor to within the icon you want to duplicate.

Step 2. Press F7 (PickIcon) to copy the icon.

Step 3. Move the cursor to fix the position where you want to place a copy of the icon. The cursor position fixes the top left corner of the duplicate icon.

Step 4. Press F6 (PutIcon) to place the copied icon at the position of the cursor.

Step 5. Attach a resource group to the icon. See the previous section, *Attaching a Resource Group to an Icon on the Icon Panel*.

Moving an Icon on the Icon Panel

Move an icon to another position on the icon panel as follows:

Step 1. Move the cursor to within the icon you want to move.

Step 2. Press F5 (CutIcon) to take the icon off the icon panel. The selected icon is no longer displayed.

Step 3. Move the cursor to fix the position of the top left corner of the icon being moved.

Step 4. Press F6 (PutIcon). The icon is redisplayed at the position of the cursor.

Editing an Icon on the Icon Panel

You can edit an icon while you are on the Icon Panel Generator panel. Editing enables you to update the original icon or create a new copy of the icon.

Updating an icon from the Icon Panel Generator panel updates the icon definition in the knowledge base and the selected icon only. If there are other icons in the panel definition that use the same icon definition, these other icons are not updated as long as you remain in the panel definition. You can, therefore, have several versions of the same icon in the panel definition. When you generate the icon panel, the panel reflects these different versions of the icon (even though there is only one version of the icon definition).

Note

Although a generated icon panel can retain different versions of the same icon, the icon panel definition cannot. The next time you access the panel definition, the definition reflects the latest version of the icon.

Edit an icon on the icon panel as follows:

Step 1. Move the cursor to within the icon you want to edit.

Step 2. Press F8 (EditIcon). The Icon Editor panel is displayed.

Caution

If you wish to cancel the operation, you must press F12 (Cancel) before Step 4.

Step 3. Edit the icon as described in the sections, *Using the Icon Editor Panel*, on page 12-8 and, *Editing the Icon*, on page 12-9. If you want to create a new copy of the icon, change the value in the Name field.

Step 4. Press F3 (File) to file the updated definition and return to the Icon Panel Generator panel.

Getting Information About an Icon on the Icon Panel

You can display the name of and the resource group attached to an icon on the icon panel. Press F10 (Query), and a message displays the required information. The following example identifies the icon as CVNEW with an attached resource group named ACREC:

```
RM810017 ICON=CVNEW RESOURCE GROUP=ACREC
```

Deleting an Icon From the Icon Panel

Delete an icon from the icon panel as follows:

Step 1. Move the cursor to within the icon you want to delete.

Step 2. Press F5 (CutIcon) to take the icon off the icon panel. The selected icon is no longer displayed.

Caution

The CutIcon action temporarily stores the icon that is taken off the icon panel. However, the icon is lost if you use the F7 (PickIcon) or F5 (CutIcon) function key on another icon.

Maintaining Icon Panel Definitions

You can browse, update, copy, and delete icon panel definitions from the Icon Panel Definition List panel.

Note

You cannot update an icon panel definition if the associated icon panel is being used by a graphical monitor.

At the end of an update or copy operation, if an icon in the panel definition does not have an attached resource group, you cannot generate the panel. A message is displayed on your screen to advise you of the fact. You can either attach any missing groups so that you can generate the panel or press F3 (File) again to file the definition without generating the panel.

Editing a Generated Icon Panel

To update an icon panel, you can either regenerate the panel by using an updated definition or you can edit the panel description file directly.

Enter the **/GADMIN.E** path to access the list of icon panels. The Panel List panel is displayed. Figure 12-9 is an example.

Figure 12-9. Panel List Panel

```
LIB: ICOPANL--------- Automation Services : Panel List -------------------EDIT
Command ===>                                                   Scroll ===> CSR

      S/U=Update B=Browse I=Information V=View P=Print D=Delete C=Copy R=Rename
    Name          Description
    $RMDEFAULT    Default icon monitor
    ACCOUNT       Accountants services.
    ACCOUNTSPAY   Accounts payable service
    ACCOUNTSREC   Accounts receive service
    BIGPICTURE    The big picture
    CCVALLFLDS    All icon fields present
    IMSMONITOR    IMS region
    TAPEMON       Tape monitor
    TAPEUNITS     Tape and cartridge units
    **END**
```

The panel displays the list of icon panels in the knowledge base. Some of these panels might be generated by using icon panel definitions; some of these panels might be created by users (for example, by using the Copy or Rename action). If an icon panel definition generates the panel, the Name and Description columns reflect the name and description of the definition.

See the *Management Services User's Guide* for information about the Edit Services that enables you to edit panel description files. See the *Network Control Language User's Guide* for information about panels and panel statements.

Note the following points when you edit an icon panel description file:

- If you regenerate an icon panel by using the P - Define Icon Panels option, you lose whatever editing you did in the description file. Use the R action to rename the panel before editing.
- The first line in a description file is the panel description, as displayed on the panel list.
- The #NOTE #ICON statement in a description file associates the specified resource groups with the icon panel.

13

Monitoring and Controlling Services and Resources

This chapter describes how to monitor and control services and resources on a day-to-day basis. The operations environment has been established with an active system image. See Chapter 14, *Controlling the System Image*, for information about starting and stopping the system image.

This chapter contains the following topics:

- Monitoring Services and Resources
- Controlling Services and Resources
- Using the Status Monitor
- Using the Graphical Monitor
- Using Commands
- Recovering a Failed Resource
- Recovering From Failed Automation
- Acknowledging a Link Failure
- Responding to the Initialization Status Panel
- Responding to the Database Synchronization Panel
- Using the Logs
- Obtaining Operations Statistics

Monitoring Services and Resources

The status and graphical monitors enable you to monitor at the service and resource levels within your operations environment rather than purely by events and the messages from resources. This reduces the amount of information displayed on the monitor screen.

You can monitor the services and resources from any of the connected focal point regions. From subordinate regions, you can monitor locally managed resources only. Logs can be used to display the messages associated with a service or resource and therefore with system activity, and authorized users can define filters and profiles that enable the viewing of information about specific services and resources.

The two monitors enable you to monitor the service and resource activities from two different viewpoints. Use the status monitor to view the status of *individual* services and resources, and the graphical monitor to view the status of *groups* of services and resources.

Controlling Services and Resources

A monitored service or resource can operate in the AUTOMATED operation mode, the MANUAL operation mode, or the IGNORED operation mode. In the AUTOMATED operation mode, the region has control of the service or resource but you can still indirectly influence the operation of that service or resource. In the MANUAL operation mode, you have direct control of the service or resource. The IGNORED operation mode is the same as the MANUAL operation mode except that the logical state of the affected service or resource is always set to OK (used typically when the state of the service or resource is of no interest).

From the monitors, you can perform the following:

- You can override the operation mode or the desired state of a service or resource, thus indirectly influencing its behavior.
- You can, if authorized, change the active knowledge base information dynamically without having to shut down the region. For example, a service level agreement might have changed and you need to update the resource availability to reflect the change.
- You can start or stop a service or resource that is in the MANUAL operation mode.

Working With Resources Registered to the Automatic Restart Manager

If a resource is registered to the automatic restart manager (ARM), its displayed mode of operation is suffixed with -ARM (for example, AUTO-ARM). The region delegates any restart processes to ARM. The region takes control only if ARM cannot complete the restart process successfully.

Using the Status Monitor

The status monitor displays services and resources line by line, and uses color and highlighting to notify you of changes in their states. The display attributes and the display format can be customized by a user with the appropriate authority. See the *Automation Services Administrator Guide* for information about the default display attributes and the display formats, and how to customize them.

If a state change occurs for a service or resource that is off the display area, a message is displayed to alert you of the change.

When service or resource status changes, you can take a number of actions depending on the situation. For example, you can access the log; override the operation mode, desired state, and actual state; display information; perform start or stop operations; and issue commands. Authorized users can also update the service and resource definitions.

The product provides the following two customized status monitors: the service status monitor (not applicable in subordinate regions), and the general resource status monitor.

Accessing the Status Monitor

From the primary menu, enter **M** to select the Monitoring : Primary Menu. From the menu, enter the following to access the appropriate variation of the status monitor:

- **S** to access the service status monitor
- **R** to access the general resource status monitor

The Monitor panel is displayed. This panel lists the entities monitored in the currently active system images. You can select a predefined filter to display specific services and resources. The customized status monitors use the following filters:

- SERVICEVIEW filter is used by the service status monitor.
- RESOURCEVIEW filter is used by the general resource status monitor if you are *not* using a default filter. The section, *Changing Your Default Status Monitor Profile*, on page 13-11 describes how to specify your default filter.

For information about status monitor filters, see Chapter 11, *Defining and Maintaining Filters for the Status Monitor*.

The format of the displayed information is determined by your status monitor profile. Figure 13-1 shows an example of a status monitor screen. The last resource is of the USRCLS class under the LOGICAL subclass.

Figure 13-1. A Status Monitor Screen

```
                    S=Status L=Transient Log D=Display DB=Database ?=List Cmds
System   Class    Resource     Desired  Actual     Mode       Logical   Ovr
SOLV     STC      CICSPROD     ACTIVE   UNKNOWN    MANUAL     UNKNOWN
SOLV     STC      SOLVCICS     ACTIVE   INACTIVE   MANUAL     ATTENTION
SOLV     STC      SOLVCIC1     INACTIVE INACTIVE   MANUAL     OK
SOLV     STC      SOLVDB2      ACTIVE   INACTIVE   MANUAL     ATTENTION
SOLV     STC      SOLVIMS      ACTIVE   ACTIVE     AUTOMATED  OK
SOLV     STC      SOLVIMS1     ACTIVE   ACTIVE     AUTOMATED  OK
SOLV     STC      SOLVIMS2     ACTIVE   ACTIVE     AUTOMATED  OK
SOLV     STC      SOLVJES      ACTIVE   ACTIVE     AUTOMATED  OK
SOLV     STC      SOLVJP       ACTIVE   FAILED     AUTOMATED  FAILED
SOLV     STC      SOLVLLA      INACTIVE INACTIVE   AUTOMATED  OK
SOLV     STC      SOLVNM       ACTIVE   FAILED     AUTOMATED  FAILED
SOLV     STC      SOLVRMF      ACTIVE   ACTIVE     AUTOMATED  OK
SOLV     STC      SOLVTSO      ACTIVE   ACTIVE     MANUAL     OK
SOLV     STC      SOLVVLF      ACTIVE   ACTIVE     AUTOMATED  OK
SOLV     STC      SOLVVPS      ACTIVE   ACTIVE     AUTOMATED  OK
SOLV     STC      SOLVVTAM     ACTIVE   ACTIVE     AUTOMATED  OK
SOLV13   LOGICAL  CONTESTADOR  INACTIVE ACTIVE     MANUAL     ATTENTION DSI
```

Interpreting the Status Monitor Display

The default status monitor display format provides the following information:

- The name of the system image that contains the service or resource. For a service, the name is $SERVICE.
- The name of the service or resource and its class. For a service, the class is SVC.

  Note

  Resources of the INTNL class with names in the form *xx*(*) are dynamic resources that provide communications between regions.

- The operation mode, desired state, actual state, and logical state.

- The override flag that indicates whether the operation mode or the desired state is overridden. Table 13-1 on page 13-6 describes the override flags.

Table 13-1. Override Flags

Override Flags	Meaning
DSA, DSI	The service or resource has a *desired state override* placed on it.
G	The operation mode of the resource is set to AUTOMATED but the mode is ineffective because the *global operation mode is MANUAL*.
M	The service or resource has an *operation mode override* placed on it.
O (CICS)	The resource has an *INACTIVE desired state override* placed on it *by its owner when the owner becomes inactive*.
P	The resource has an *INACTIVE desired state override* placed on it *by a parent's scheduled availability*.
SVC	The service or resource has an *ACTIVE desired state override* placed on it *by the starting of a host service* (that is, a service that requires it to be active).
T	The service or resource has an *INACTIVE desired state override* placed on it *by the issuance of a TRI command on a parent*.
TRI	The resource has an *INACTIVE desired state override* placed on it *by a TRI command*.
G/D, G/O (CICS), G/P, G/S, G/T, M/D, M/O, M/P, M/S, M/T	These flags indicate *combined operation mode and desired state overrides*, where DSA and DSI are abbreviated as D, and SVC is abbreviated as S.

A blinking service or resource indicates that the region is performing an automated action on it.

You are notified of changes in the status of monitored items that are not in view.

Operating in the Status Monitor

From the status monitor, you can issue commands to solicit information about, or to modify the behavior of, a service or resource. You can use the Z command to zoom in to a service to find out the status of its members. The section, *Using Commands*, on page 13-16 contains detailed information about some of the tasks you can perform by using commands. For descriptions of the supplied commands, see help.

Organizing the Information on the Status Monitor

You can enter the following commands at the Command ===> prompt to organize the information on the status monitor.

COLS {1|2|3|4|5}

Use this command to display the services and resources in up to five columns across your screen. For example, enter **COLS 5** to display in the 5-column format. This command is useful if there is more than one panel of services and resources.

This command does not apply to some status monitors with views. For information about views, see *Status Monitor Views*, on page 11-4.

EXTDISP {OFF|ON}

Use this command to control the display of user-defined extended displays. User-defined extended displays, if they exist, overwrite the last four columns of the normal status monitor display.

For example, the status line for a tape unit might contain a text string that provides information about a tape mounting operation.

FORMAT {?|*format-name*}

Use this command to change the way information is displayed on your screen for a particular columns setting. If you do not know the name of a format, enter **FORMAT** and select it from the displayed list of format definitions.

Note

The product supplies a number of predefined formats. If authorized, you can define your own formats from the List Definition List panel. To access the list, enter the **/ASADMIN.L** path. See the *Automation Services Administrator Guide* for information about how to define status monitor formats.

If you updated the current format, you need to enter the REFORMAT command to make the updates effective in the current session.

SORT *operand*

Use this command to sort the services and resources displayed on your screen in a particular order. *operand* can be one of the following:

?

displays a list of available operands to use with the SORT command. Enter the number of the operand. The sort is in the order you specified. See the following SORT operands for a description of the types of sorts.

ACTSTAT

enables you to sort the services or resources by the actual state according to the following in descending order of ranking: UNKNOWN, FAILED, INACTIVE, DEGRADED, STOPPING, STARTING, and ACTIVE.

CLASS

enables you to sort the services and resources by the class number in numeric order. See Appendix H, *Knowledge Base Definition Classes*, for the list of class numbers.

CLSNAME

enables you to sort the services and resources by the class name in alphabetic order.

DSTACUR

enables you to sort the services and resources by the desired state according to the following in descending order of ranking: ACTIVE, INACTIVE, and UNKNOWN.

LOGSTAT

enables you to sort the services and resources by the logical state according to the ranking in the logical state attributes table.

MODECUR

enables you to sort the services and resources by the operation mode according to the following in descending order of ranking: IGNORED, MANUAL and AUTOMATED.

NAME

enables you to sort the services and resources by the name in alphanumeric order.

OBJID

enables you to sort the services and resources by the object ID that comprises (in order) the system image name, the system image version, the class number, and the name.

SYSNAME
: enables you to sort the services and resources by the system image name in alphabetic order.

TYPE
: enables you to sort the resources by the resource type in alphabetic order. Services, which do not have the TYPE parameter, are listed at the beginning.

USRKEYW
: enables you to sort the resources by user-defined filter keywords. Services and resources for which filter keywords are currently not applicable are listed at the end.

  Filter keywords are defined in the resource definitions. The keywords are message related. That is, a keyword becomes active only if certain messages are received for a resource.

  For example, if a keyword is specified in the rule for the IEC501A message that requests a tape mounting operation, you can use the USRKEYW operand to group tape units that request tape mounting operations together on the status monitor.

Note

Services and resources that rank the same by *operand* are further sorted by the object IDs.

Changing to a Different Status Monitor Filter

Authorized users can create filters to define which services and resources are displayed on the status monitor. Filters are sets of rules stored in the knowledge base that enable you to display a subset of the monitored services and resources. For example, a tape operator may only want to see the tape units in each system, or in a combination of systems. Another operator might want to see only the batch jobs in the production system during his or her shift. A third operator may want to see all the hardware resources for a particular system.

The services and resources you see the first time you access the status monitor depend on the filter specified in your user profile. If no filter is specified, you see all the services and resources. You can enter the following primary commands at the Command ===> prompt to select a different status monitor filter:

FILTER	Enter this command to display a list of the available filters. Enter the number that identifies the filter you want to use. Only the services and resources that match the filter are displayed on the status monitor.
FILTER *filter-name*	Enter the name of the filter that you want to use. Only the services and resources that match the filter are displayed on the status monitor.
FILTER NONE	This command removes any filtering. You now see every service and resource for the systems to which you are connected.

Note

If authorized, you can enter **F** on the Automation Services Administration Menu panel to select the Status Monitor Filters option to create and maintain status monitor filters. You can also use the **/ASADMIN.F** path to select the Status Monitor Filters option.

If you updated the current filter, you need to enter the REFILTER command to make the updates effective in the current session.

Changing Your Default Status Monitor Profile

When you access the status monitor, the type of information available is determined by your default status monitor profile. The profile specifies the following:

- Filter to use when you first enter the general resource status monitor. The filter is not applicable to the other status monitors.
- Number of monitored entities to display across the screen and the corresponding display format.
- Criteria that determine how the monitored entities are sorted for display.
- Flag that indicates whether to process user-defined extended display.

To change the defaults, enter **PROFILE** at the Command ===> prompt. A Profile panel is displayed. This panel contains fields that enable you to change your default profile for the status monitor.

To select a new default, enter **?** in the appropriate field. A list of available values is displayed. Select a value, and press ENTER.

When you finish with the Profile panel, do one of the following:

- Press F3 (File) to save the new defaults and return to the status monitor.
- Press F4 (Save) to save the new defaults and remain on the panel.
- If you change your mind and do *not* want to change the defaults, press F12 (Cancel) to return to the status monitor without saving the changes.

Using the Graphical Monitor

The graphical monitor displays groups of services and resources as graphical images called *icons*. This monitor uses display attributes to alert you to changes in the status of the represented groups. Changes in an icon are caused by changes in the members in the group. Thus, if an icon changes color, you can look at its members to see what is causing the change.

The display attributes of an icon reflect the status of the represented member that is in the worst state. The display attributes for icons are controlled by the display attribute tables, except that icons do not use the highlighting attribute, as all icons appear in reverse video. The display attributes can be customized by a user with the appropriate authority. See the *Automation Services Administrator Guide* for information about the default display attributes and how to change them.

An icon is associated with a group of services and resources, or a panel, as follows:

- Group of services and resources—a group, when represented by an icon, enables items managed by the region to be presented with a service-driven operations perspective.

  When you select an icon associated with a group, the members in the group are listed. You can then monitor the individual services and resources.

  When you issue an operations command against an icon associated with a group, the command acts on all the members.

- Panel—when you select an icon associated with a panel, a panel of component icons is displayed. For example, an icon may represent all the resources in the system EASTTEST. When you select the icon, a panel of icons is displayed. The component icons may contain specific types of resources (for example, all the initiators in EASTTEST, all the started tasks in EASTTEST, and so on). Each of these icons may be associated with another panel or with a group of resources. You can follow a very specific path leading directly to any resources in the system that are causing the problem.

  When you select an icon associated with a panel, a panel of component icons is displayed. For example, an icon may represent all the resources in the VTAM domain EASTTEST. When you select the icon, a panel of icons is displayed. The component icons may contain specific types of resources. Each of these icons may be associated with another panel or with a group of resources. You can follow a very specific path leading directly to any resources in the system that are causing the problem.

  You cannot issue operations commands against an icon associated with a panel.

Accessing the Graphical Monitor

From the primary menu, enter the **/GMON** shortcut to select the graphical monitor option. The Graphical Monitor panel is displayed.

Figure 13-2 shows an example of a graphical monitor screen.

Figure 13-2. The Graphical Monitor Screen

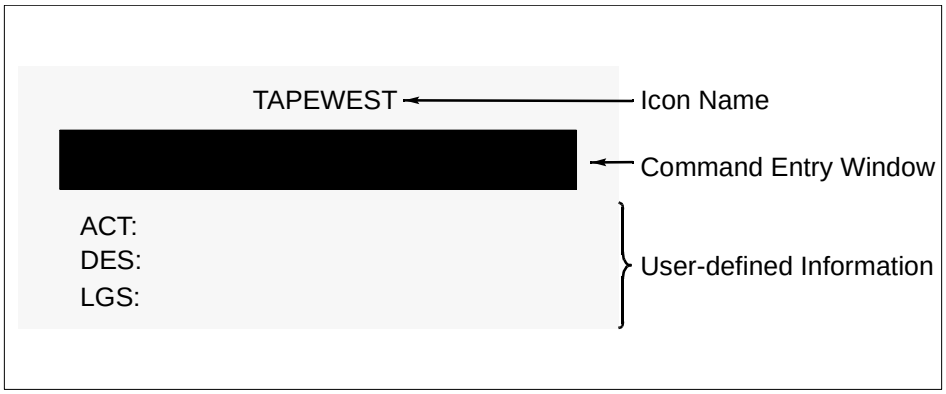
```
SOLVPROD------------------------ Graphical Monitor ---------------------ACCOUNT
Command ===>

                              SOLV1
                              DSC:MAIN SOLV1 ICON

                              ACT:INACTIVE
                               LGS:FAILED

                              SOLV2
                              DSC:MAIN SOLV2 ICON

                              ACT:INACTIVE
                               LGS:UNKNOWN

 F1=Help      F2=Split     F3=Exit      F4=Return    F5=Panel
                           F9=Swap
```

Interpreting the Graphical Monitor Display

Icons on the graphical monitor group services and resources together (for example, all tape units, all started tasks, or all the resources required for a particular system). Icons can group different types of resources, and can group resources from different systems.

Each icon is identified by a name. You can enter commands in the command entry window. Figure 13-3 shows an icon and the information it contains.

Figure 13-3. An Icon



An icon may display other information (for example, the resource states).

Operating From the Graphical Monitor

From the graphical monitor, you can:

- Use the Z (Zoom) command to view the status of the services and resources attached to an icon
- Issue commands to solicit information about, or to modify the behavior of, all the services and resources attached to an icon

Using the Z (Zoom) Command

Use the graphical monitor to get a high level view of the monitored services and resources. If an icon indicates a failure, issue the **Z** command from the icon command entry window to access the icon components and find the problem.

Zooming to the Status Monitor

If the icon is associated directly with a list of services and resources, the Monitor panel is displayed. This panel contains only those services and resources defined to the selected icon. For example, if you zoom from a group called TAPEWEST, which contains tape units for the WESTPROD system, the monitor panel only contains information about tape units in the WESTPROD system.

Use the Monitor panel to check the status of resources to see which resources are causing the problem. You can then perform status monitor actions, access the logs, or issue line commands to locate the source of the problem.

Zooming to an Icon Panel

If the icon is associated with a panel, the panel is displayed. This panel contains icons that are subsets of the main icon. For example, the icon containing all resources in EASTTEST might change to red, indicating a problem. You can zoom to the next panel of icons to see which icon has caused the main icon to change color. Select any icons that have changed state to see which resources are causing the problem. Once you reach the status monitor, you see the status of the individual resources.

Using Other Commands

You can enter line commands in the command entry field of an icon that directly represents a group of services and resources (that is, an icon that is *not* pointing to another icon panel). To list the available commands, enter **?** in the field.

A command acts on all the members of the group. For example, if you issue the A command against the TAPEWEST icon, you start all inactive TAPEWEST tape units that are in the MANUAL operation mode.

When you use a command on a group, a Confirm Command panel might appear (depending on a setting in the DISPLAYS region parameter group). See the *Automation Services Administrator Guide*.

The panel advises you of the number of members in the group. You can:

- Press F6 (Action) to confirm the command.
- Press F5 (Zoom) to list the members if you decide that you do not want to issue the command against all the members. The list enables you to issue the command against individual members.
- Press F12 (Cancel) if you change your mind and do not want to issue the command.

Changing to a Different Default Icon Panel

When you access the graphical monitor, the type of information available is determined by your default icon panel. To change the panel, enter **PROFILE** at the Command ===> prompt. A Profile panel is displayed. This panel contains a field that enables you to change your default icon panel for the monitor.

To select a new default, enter **?** in the Panel Name field. A list of available panels is displayed. Select the required panel by entering **S** beside the panel name.

When you finish with the Profile panel, do one of the following:

- Press F3 (File) to save the new default and return to the graphical monitor.
- Press F4 (Save) to save the new default and remain on the panel.
- If you change your mind and do *not* want to change the default, press F12 (Cancel) to return to the graphical monitor without saving the changes.

Note

If you want to view a different icon panel temporarily only, use the F5 (Panel) function key or the PANEL command.

Using Commands

You can issue commands to perform various actions from the status and graphical monitors, depending on your authority level. This section discusses some of the common actions.

Use the **?** command to find out what commands are available for the selected service or resource. For descriptions of the commands, see help.

Finding Out More About Monitored Services and Resources

Use the **D** command to display information about the selected services or resources.

Use the **B** or **S** command to display information about the current and defined states, and the operation mode of selected services and resources. If necessary, you can override the current operation mode, actual state, and desired state.

Note

You can also use commands to override the operation mode and the states. The section, *Overriding the Operation Mode*, on page 13-18 and the section, *Overriding the Service and Resource States*, on page 13-19 describe the override commands.

Finding Out About Relationships Between Monitored Resources

You might want to determine the resource relationships when, for example, you are going to perform an action that may affect the actual state of a resource that may in turn affect related resources. You can display resource relationships in one of the following ways:

- To view all relationships in the local system image, enter **GR** or **GRT** at the Command ===> prompt.
- To view the immediate parents and children of a resource, enter **GR** (graphical display) or **R** (line by line display) beside the resource.
- To view all the children of a resource, enter **GRT** beside the resource.

If you use the GR command beside a resource, you can Press the F4 (Explode) key to view all the relationships in the system image.

If you use the GR or GRT command, you can also print the displayed relationship by entering the PRINT command.

Removing the User-defined Extended Display

If the last four columns of a full width status display are overwritten by a user-defined extended display and you want to view the original information in those columns, use the **EDR** command. The command removes the extended display temporarily. The extended display reappears on the next update of the status.

Note

Enter the EXTDISP {OFF|ON} command at the Command ===> prompt to turn the extended display on or off for all the resources. The setting stays for the duration of the session.

If you want to retain the setting across sessions, specify the setting in your status monitor profile. The section, *Changing Your Default Status Monitor Profile*, on page 13-11 describes how to change the settings in your profile.

Accessing Extended Functions

Use the **XF** command to access any extended functions provided for selected services and resources.

Replying to a WTOR Message

When the status of a resource or service indicates that there are outstanding WTOR messages waiting to be replied to, Enter **W** beside the resource or service. A Command Entry panel is displayed listing the WTOR messages.

To reply to a WTOR message, issue the following command at the SYSCMD ===> prompt:

`REPLY` *`wtor-id,reply-text`*

Overriding the Operation Mode

The following commands enable you to modify the operation mode of selected services and resources:

- Use the **MA** command to set the operation mode to AUTOMATED. The region takes control of the affected service or resource.

  Note

  If the global operation mode is MANUAL, the mode cannot change even though the AUTOMATED override is applied. This condition is indicated by a G in the Ovr column. The section, *Changing the Global Operation Mode*, on page 14-2 describes the global operation mode.

- Use the **MM** command to set the operation mode to MANUAL. The region relinquishes control of but continue to monitor the affected service or resource.
- Use the **MI** command to set the operation mode to IGNORED. This mode is the same as the MANUAL mode except that the logical state of the affected service or resource is always set to OK. Use this mode if you are not interested in the state of the service or resource.
- Use the **MR** command to remove the override.

On the status monitor, an M in the Ovr column identifies a service or resource as having an operation mode override.

Overriding the Service and Resource States

You can override the actual state and the desired state of selected services and resources.

Overriding the Actual State

Override the actual state only if you believe that the displayed state is wrong. The following commands enable you to modify the actual state of selected services and resources:

- Use the **ASA** command to set the actual state to ACTIVE.
- Use the **ASD** command to set the actual state to DEGRADED.
- Use the **ASF** command to set the actual state to FAILED.
- Use the **ASI** command to set the actual state to INACTIVE.
- Use the **ASU** command to set the actual state to UNKNOWN to recheck the status.

Overriding the Desired State

The following commands enable you to modify the desired state of selected services and resources:

- Use the **DSA** command to set the desired state to ACTIVE.
- Use the **DSI** command to set the desired state to INACTIVE. If you enter DSI beside a resource or service that has the SVC override flag, a panel is displayed listing the services that might be affected by the action.
- Use the **DSR** command to remove the override.

On the status monitor, a DSA, DSI, G/D, or M/D, in the Ovr column identifies a service or resource as having a desired state override.

If you set the desired state of a service to ACTIVE, then when the service is started (either through automation or by you manually), the service places an ACTIVE desired state override on its members. On the status monitor, an SVC in the Ovr column identifies a service or resource as having a desired state override imposed on it by a service. Similarly, if you set the desired state of a service to INACTIVE, then when the service is stopped, the service removes the ACTIVE desired state override from its members.

Stopping a Parent and Its Children

You can stop selected active resources and their children that are in the AUTOMATED operation mode. Use the **TRI** command to override the desired states of the resource and its children (that is, members of the relationship tree) with the value INACTIVE. Confirmation panels enable you to find out what services might be affected by the action. When you confirm the action, automation stops the automated children to bring their actual states to match the INACTIVE desired states.

Note

When you issue the TRI command against a resource, the Ovr column on the status monitor has the value TRI for that resource, and the value T for its children. To put the resource and its children back to the original desired states, use the TRR command.

If one of the children is in the MANUAL or IGNORED operation mode and is active, then that child and its parents are *not* stopped (even if the parents are in the AUTOMATED operation mode).

Browsing or Updating the Definition of Monitored Services and Resources

Use the **DB** command to browse selected service and resource definitions. Users with the appropriate authority can also update the definitions. The update is effective immediately. For example, you may need to add a new status monitor message.

Caution

Resources of the INTNL class with names in the forms DT(*), ET(*), and ST(*) are dynamic APPC resources that provide communications between regions. These resources are defined only for the duration of the APPC link. Some products also use dynamic resources of the form *xx*(*).

Do *not* change these resource definitions.

Note

You can use the F4 (Add) function key or the ADD primary command to add definitions to and the DEL command to delete definitions from the knowledge base. For more information, see help.

Modifying the Scheduled Operations Changes for a Monitored Item

Scheduled changes in the operations of a service or resource is defined in an availability map. An availability map defines the desired service or resource status at specific times. If a service or resource is attached to the map and is in the AUTOMATED operation mode, the region will try to ensure that the actual status of the service or resource matches the desired status defined in the map. More than one service or resource can use the same map.

You can modify those changes in one of the following ways:

- Create a map to override any existing requirements.
- Update the attached map. This action, however, also modifies the behavior of other services and resources attached to the map.

Overriding the Scheduled Operations Changes

Use an overriding map when you want to modify the operations behavior of a service or resource for a defined period of time only.

Enter **MPO** to create an overriding availability map that specifies your new requirements. You must specify the map expiry date and time, and the map is deleted automatically at the specified date and time. The scheduled operations changes then revert back to what were specified originally.

You can revert the scheduled operations changes back to what were specified originally, at any time. To do this, enter **MPR**. The command replaces the overriding map by the original map. The command does not, however, delete the overriding map.

Once the map is created, it is available for use by other services and resources. (To attach an overriding map to another service or resource, specify the name of the map in the appropriate definition. You can update the definition by using the DB command from your monitor or through the Definition menus.)

Browsing or Updating Availability Maps

Caution

Updating an availability map affects the operation of all the services or resources attached to the map.

Use the **MAP** command to browse selected service and resource definitions. Users with the appropriate authority can use the **MAP** command to update availability maps to which selected services and resources are attached. The update is effective immediately.

Moving the Control of a Resource From One Region to Another

Note

If your region has the SOLVE:Operations Automation product that supports shared system images, you should define resources that can become active on different systems in those images. For information about shared images, see the *Unicenter SOLVE:Operations Automation User's Guide*.

Use the **MV** command to move the control of a resource from the region on one system to the region on another system.

In a multisystem environment, you might want to balance the load on the systems by moving the control of certain shared resources from one system to another. Those resources should have been defined to the affected system images already, together with the relevant resource relationships, availability map, and processes. The defined resources are controlled by one system image at any one time. The definitions in the other images are set to the OFF operation mode (that is, not displayed on the monitors). During a move operation, the following actions occur:

1. The operation mode of the definition in the current system image is set to MANUAL.
2. The resource is stopped or inactivated in the current system.
3. The operation mode of the definition is set to OFF, removing the resource from the monitors. The control of the resource is removed from the current system.
4. The operation mode of the definition in the target system image is set to the value before Step 1. The status of the resource is redisplayed under the target system image. The control of the resource is moved to the target system.

Note

Resource relationships do not cross system image boundaries. If a number of resources are in a relationship and you want to move the relationship, you must move all those shared resources. For the operation to be successful, the system image to which the resources are moved must have a duplicate relationship defined.

Controlling Manual Services and Resources

You have direct control over services and resources that are in the MANUAL or IGNORED operation mode. That is, you can start and stop manual services and resources. You can use these modes of operation when automatic error recovery fails.

Note

For a resource, these operations use the methods specified in the definition for that resource, provided that you have not registered commands that override the supplied commands.

Starting Manual Services and Resources

You can start selected services and resources manually. For example, you may want to restart a resource after recovering it from a failure. Ensure that the operation mode of the service or resource you want to start is MANUAL or IGNORED. Use the **A** command to start the services and resources. A message is displayed on your screen prompting you for confirmation. Press ENTER to confirm the action. The actual state changes to ACTIVE if the service or resource starts successfully.

Stopping Manual Services and Resources Normally

You can stop selected services and resources manually. For example, you might want to stop a resource because its status is degrading and you want to take it out of service. Ensure that the operation mode of the service or resource you want to stop is MANUAL or IGNORED. Use the **T** command to stop the services and resources. A message is displayed on your screen prompting you for confirmation. Press ENTER to confirm the action. The actual state changes to INACTIVE if the service or resource stops successfully.

Stopping Manual Resources by Force

You can stop selected resources manually by force. For example, in an emergency, you want to stop a resource quickly. The normal stop process can take some time to complete. Ensure that the operation mode of the resource you want to stop is MANUAL or IGNORED. Use the **TF** command to stop the resources. A message is displayed on your screen prompting you for confirmation. Press ENTER to confirm the action. The actual state changes to INACTIVE when the resource has stopped.

Recovering a Failed Resource

If a resource fails and no automated recovery action is performed, the resource remains in the FAILED actual state, and you need to recover the resource manually. Proceed as follows:

Step 1. Ensure that the resource is in the MANUAL operation mode. If the mode is AUTOMATED, enter **MM** beside the resource to set its operation mode to MANUAL. This indicates to other operators that the resource is being attended to.

Step 2. Enter **L** beside the resource to access its transient log. The log provides information about the cause of the problem, which helps you determine the actions to take.

Step 3. Correct the problem, and ensure that the resource can be safely restarted.

Note

You can use the Command Entry facility to issue system commands. Enter **CMD** at the Command ===> prompt, and issue the SYSCMD command from the Command Entry panel. See the *Command Reference* for information about the SYSCMD command.

Step 4. Enter **CHK** or **RES** beside the resource to reset the status of the resource. If the resource is in the AUTOMATED operation mode, automation restarts for the resource. If the resource is in the MANUAL operation mode, enter **A** beside the resource to start it.

For the differences between the CHK and RES commands, see help.

Recovering From Failed Automation

Automation fails for a resource in the AUTOMATED operation mode when the region cannot maintain the desired state of the resource. The status monitor displays the resource as having an INERROR logical state. To recover from the problem, proceed as follows:

Step 1. Enter **MM** beside the resource to set its operation mode to MANUAL. This indicates to other operators that the resource is being attended to.

Step 2. To confirm that automation has actually failed, enter **S** beside the resource to access the Modes and States panel. The panel displays the automation status, which should be FAILED.

Step 3. Press F12 (Cancel) to return to the status monitor. Enter **L** beside the resource to access its transient log. The log provides information about the cause of the problem, which helps you determine the actions to take. A likely cause is that the recovery action specified in the resource definition is inappropriate.

Step 4. Correct the problem, and ensure that the resource can be safely restarted.

Step 5. Enter **RES** beside the resource to reset the status of the resource. Automation then restarts for the resource.

Acknowledging a Link Failure

When a link between connected regions fails, the resources being monitored through the failed link appear in the UNKNOWN state. A failed link can affect a large number of resources and fill your status monitor with link failure error states. Use the **ACKLNKFAIL** command to acknowledge that you note the failure and to clear your status monitor of the affected resources so that you can better monitor the other resources. The affected resources reappear when the link recovers.

Use the ACKLNKFAIL command as follows:

Step 1. On your Monitor panel, type **ACKLNKFAIL** at the Command ===> prompt and press ENTER. The Execute ACKLNKFAIL Command panel is displayed. Figure 13-4 is an example.

Figure 13-4. Execute ACKLNKFAIL Command Panel

```
SOLVPROD----- Automation Services : Execute ACKLNKFAIL Command ----------------
Command ===>                                                   Scroll ===> 10

+-----------------------------------------------------------------------------+
|                                                                             |
|     Select one or more of the ServiceView System Images listed below        |
|     which are to have the ACKLNKFAIL command actioned against them.         |
|                                                                             |
+-----------------------------------------------------------------------------+
+- Active System Images -----------------------------------------------------+
|                                                                S=Select Image |
|     ACB Name SysName  Version Description                                   |
|     SOLV12   MEL1     0001    Melbourne production                          |
|     SOLV14   SYD1     0001    Sydney production                             |
|     **END**                                                                 |
|                                                                             |
|                                                                             |
|                                                                             |
|                                                                             |
|                                                                             |
|                                                                             |
|  F1=Help     F2=Split    F3=Exit     F4=Return   F5=Find     F6=Refresh     |
|  F7=Bkwd     F8=Forward  F9=Swap                            F12=Max         |
+-----------------------------------------------------------------------------+
```

The panel displays the list of regions connected to your region. The panel identifies the access method control block (ACB) name of each region and the system image that is active in that region.

Step 2. Type **S** beside the regions affected by the failed link, and press ENTER. A confirmation panel is displayed.

Step 3. Enter **CONFIRM** in the Response field to execute the command. The Monitor panel is redisplayed with the resources under the control of the selected regions removed from the list of monitored resources.

Responding to the Initialization Status Panel

If the loading of a system image starts in your region while you are using the status or the graphical monitor, your monitor session ends and the Initialization Status panel is displayed. Similarly, if you attempt to access the status or the graphical monitor while the system image is being loaded, the Initialization Status panel is displayed. You can take one of the following actions:

- Press F6 (Action) to monitor the loading process.
- Press ENTER to refresh the information in the status window.
- Press F3 (Exit) to exit the panel.

You can return to the monitor when the loading process completes.

Responding to the Database Synchronization Panel

If knowledge base synchronization is started for your region while you are using the status or the graphical monitor, your monitor session ends and the Database Synchronization panel is displayed. Similarly, if you attempt to access the status or the graphical monitor while the knowledge base is being synchronized, the Database Synchronization panel is displayed. You can take one of the following actions:

- Press F6 (Action) to monitor the synchronization process.
- Press ENTER to refresh the information in the status window.
- Press F3 (Exit) to exit the panel.

You can return to the monitor when the synchronization process completes.

Using the Logs

A region keeps two types of logs: transient logs and activity logs.

Each service or resource has its own transient log. The transient log provides a real-time view of the activity associated with a particular service or resource. Transient logs are not kept between region restarts. However, depending on the log details defined for a service or resource, the transient log information may be written to the permanent activity log.

The activity log provides a historical view of the system activities associated with all the services and resources monitored by the region. A region can have more than one activity log, of which only one is open for logging. The activity log is stored on DASD, so that users can access this information if necessary. See the next section, *Accessing an Activity Log*. You can also access the activity log from the transient log. See the section, *Accessing a Transient Log*, on page 13-29.

The information that is logged is determined by the log parameters specified in the service or resource definition. The region logs the information to the transient log, as well as to any other destinations specified in the definition.

Accessing an Activity Log

Access an activity log from the Historical Data : Primary Menu.

Select option **H** on the primary menu to display the Historical Data : Primary Menu, then select option **L**. The activity log appears on your screen displaying the last logged activities. The log also displays the day, date, and time when the region was last initialized.

If you want to access the log in a linked region, identify the region in the Link Name field on the Historical Data : Primary Menu.

Browsing the Activity Log Online

The following aids are available to help you use the activity log:

- Forward and backward scroll
- Various display formats
- Highlighting log records
- Log filter to restrict the records displayed
- Positioning by date or time, either absolute or relative
- Positioning by labels set in the log
- Searching forward or backward for text
- Selective printing of log records

See the online help for detailed information about these aids.

Accessing a Transient Log

Access the transient log from the status or the graphical monitor. This log and the activity log contain messages and other information associated with a service or resource. The logs help you diagnose problems (for example, resource failure). In the following example, you want to find out about the activities of a resource called CICSA, which is in error.

Enter **L** beside the CICSA resource to access its transient log. The transient log contains a message indicating that a request to start CICSA failed. Enter **L** beside the message to access the activity log to see what was happening in the system at that time.

Messages in the transient log appear in different colors according to the severity level of the message (see Table 13-2 in the next section).

Setting Criteria to Display Logged Messages Selectively

When you access the transient log of a resource, you are profiled to view all messages logged for that resource. You can limit the number of messages you see in the current session by profiling for certain criteria.

To set profile values, enter the **PROFILE** command at the Command ===> prompt. A panel appears displaying the current profile setting. Change the settings as required

You can also supply the profile values directly by using the command operands. The syntax of the PROFILE command is:

```
PROFILE        [[SEV={1|2|3|4|5|6}]
                [NCLID=ncl-process-identifier]
                [PROC=ncl-procedure-name]] |
               [RESET=YES]
```

You can profile messages according to the following:

- Severity—determines how critical the messages are. For example, specify **SEV=1** to see only the most critical messages or **SEV=4** to see the messages of severity levels 1 through 4. Table 13-2 lists the severity levels of the messages and the color of those messages.

  Table 13-2. Severity Levels of the Messages in the Transient Log

Severity Level	Description	Color
1	Severe error	Red
2	System command issued for the resource	White
3	Change to status	Yellow
4	Warning or message that requires attention	Turquoise
5	Information (for example, responses from issued commands) or user-defined log message	Blue
6	Audit trail	Pink

- NCL process identifier—specifies the ID of the NCL process that issued the messages you want to see. The NCL processes are created when NCL procedures are executed. The region creates NCL processes when performing automation tasks.

  Each NCL process has a unique ID, so you can use the NCLID operand to differentiate between executing copies of the same NCL procedure if, for example, the same NCL procedure was run more than once in a given period.

  The region processes services and resources asynchronously, and a number of NCL processes for the service or resource might be active at the same time. Messages raised by these NCL processes intermingle in the transient log. You can use the NCLID operand to view the messages raised by a particular NCL process (that is, to view the messages for a particular task done).

- Procedure name—specifies the name of the executing NCL procedure that issued the messages you want to see.

  You can use the asterisk (*) wildcard character to include more than one NCL procedure. The wild card represents any single character except at the end of a string when the wild card represents one or more characters.

- Reset—specify **RESET=YES** to reset the profile to view all available messages.

Displaying User-defined Log Messages

A user-defined log message can be specified for a resource on the Define Event Related Actions panel when the resource is defined.

If you want to see only these messages in the transient logs, enter **UL** beside the resource.

Obtaining Help on a Logged Message

To display the detailed help text associated with a message in the transient log, enter **H** beside the message.

Printing a Transient Log

Enter **P** beside a resource to print its transient log. The PSM : Confirm Printer panel is displayed. Ensure that the displayed information is satisfactory, then press F6 (Confirm) to print the log report.

Resetting a Transient Log

Enter **LR** beside a resource to reset its transient log. This command clears the transient log of all messages.

Changing Logging Options

Logging options for a service or resource are specified on the Automation Log Details panel when the service or resource is defined. These options determine how and what information is logged. See the section, *Defining the Logging Details*, on page 5-29 for information about these options.

Merging Transient Logs

For a service, instead of displaying only the messages logged for that service, you can display a merged transient log that also includes the messages logged for the resources belonging to it. To identify the owner of a message, the merged log contains the following additional information: the system image name and version, the class name, and the service or resource name.

To display a merged transient log, enter **LC** beside the service. To refresh the log, press F6 (Refresh). The log display extends over several screens. Use the F10 (Left) and F11 (Right) function keys to scroll through the screens.

To print a merged transient log, enter **PC** beside the service.

Transient Logs From Different Time Zones

When transient logs from different time zones are merged, all dates and times refer to the local time zone.

Size of a Merged Transient Log

The size of the merged transient log is the same as the size of the transient log of the selected service. When the merged log becomes full, the oldest messages are displaced by the new messages.

Obtaining Operations Statistics

You can obtain operations statistics of a service or resource in your region or a connected region on your screen.

Obtaining the Operations Statistics of a Monitored Service or Resource

The operations statistics of a service or resource provide information about the message and command traffic in relation to the service or resource, and its actual availability. Use the ST command to display the statistics. You can display the statistics from the status monitor or the graphical monitor.

The following procedure shows you how to display the statistics from the status monitor:

Step 1. Access a status monitor. The Monitor panel is displayed.

Step 2. Enter **ST** beside the service or resource for which you want to display the statistics. The Statistics panel is displayed. See the next section, *Interpreting the Statistics*, for a description of the information on the panel.

Use F6 (Refresh) to refresh the statistics.

Step 3. Press F3 (Exit) to exit the panel when you finish browsing the statistics.

Interpreting the Statistics

The Statistics panel contains the operations statistics of a service or resource. Figure 13-5 is an example.

Figure 13-5. Statistics Panel

```
SOLVPROD------------ Automation Services : SOLVCICS Statistics -------SOLV-0001
Command ===>                                                     Function=BROWSE

 Statistics Collection commenced at 09.54.40 on MON 25-SEP-2000

+ Messages and Commands  -----------------------------------------------------+
|                                                                              |
|  Messages Processed by Automation Services ...........     15               |
|  Automation Commands Issued by Automation Services ...      1               |
|                                                                              |
+------------------------------------------------------------------------------+
+ Availability  ---------------------------------------------------------------+
|                                                                              |
|  SOLVCICS           has been UNAVAILABLE since 09.05.51 on FRI 30-MAY-1997  |
|                                                                              |
|  Availability Percentage ......................   15.4 %                    |
|  Total Time Available .........................    579 minutes              |
|  Total Time Unavailable .......................   3180 minutes              |
|  Number of times available ....................      3                      |
|  Number of times unavailable ..................      3                      |
|                                                                              |
+------------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=Exit                                F6=Refresh
 F9=Swap
```

The panel displays the time at which the statistics collection starts. There are two windows. The windows contain statistics on messages and commands, and on the availability of the selected item.

Interpreting the Messages and Commands Statistics

The Messages and Commands window contains the following statistics:

- Messages Processed provides a count of the messages processed by the region on behalf of the service or resource.
- Automation Commands Issued provides a count of the start and stop commands issued by the region on behalf of the service or resource.

If the service or resource is available, the Availability window displays the time at which the service or resource became available; if the service or resource is *not* available, the window displays the time at which the service or resource became unavailable. The window contains the following statistics:

- Availability Percentage indicates the portion of time when the service or resource is available as a percentage.
- Total Time Available gives the number of time units over which the service or resource is available.
- Total Time Unavailable gives the number of time units over which the service or resource is unavailable.
- Number of Times Available gives the number of times the status of the service or resource goes from unavailable to available.
- Number of Times Unavailable gives the number of times the status of the service or resource goes from available to unavailable.

14

Controlling the System Image

This chapter discusses the commands that authorized users can use from the status and graphical monitors to control an active system image.

This chapter contains the following topics:

- Changing the Global Operation Mode
- Loading a System Image
- Shutting Down Resources in an Active System Image
- Restarting Resources in an Active System Image

Chapter 13, *Monitoring and Controlling Services and Resources*, describes how to control individual resources and SNA groups.

Changing the Global Operation Mode

A region runs in a global operation mode of MANUAL or AUTOMATED, with the latter having a higher rank. The global operation mode is the highest mode that any resource can run in. For example, if the global operation mode is MANUAL and the operation mode of a resource is AUTOMATED, the resource can only run in the MANUAL operation mode, unless the global operation mode is changed to AUTOMATED.

You can issue a GLOBAL command from the monitor to set the global operation mode in which the resources controlled by that region can run. For example, when you have finished testing a system image in a development system in the MANUAL operation mode, you might want to change the global operation mode to AUTOMATED. If you are experiencing severe problems in a production system, you might want to change the global operation mode from AUTOMATED to MANUAL.

The syntax is as follows:

```
GLOBAL      MODE={AUTOMATED | MANUAL}
            [SYSNAME=systemname] [VERSION=nnnn]
```

MODE= Specifies the global operation mode for the system image. Table 14-1 describes the effect of the global operation mode on the resources managed by the region.

SYSNAME= Specifies the name of the local system image that is active in the region for which you want to change the global operation mode.

VERSION= Specifies the version of the local system image that is active in the region for which you want to change the global operation mode. You must specify a 4-digit value (for example, 0001).

Table 14-1. Global Operation Modes

Global Operation Mode	Effect on Resource Operation Mode
AUTOMATED	As the AUTOMATED operation mode is the highest level of region control, all resources operate in their normal modes as specified in the knowledge base.
MANUAL	Resource operation modes (except those of the INTNL class resource, which must always be automated) are set to MANUAL, if they are not already operating in the MANUAL operation mode, until the global operation mode is changed. The modes specified in the resource definitions are not changed.

Example

If the region is running in the MANUAL operation mode and you want to test the effects of automation on the resources in the system, set the global operation mode to AUTOMATED.

Enter the following command at the Command ===> prompt of the monitor:

```
GLOBAL MODE=AUTOMATED
```

The Execute GLOBAL Command panel is displayed. Enter **S** beside the required system image. The region sets all the resource operation modes to their normal value.

Loading a System Image

The region loads a user-specified system image during region initialization. During operation, you may need to change the system image by loading another image (for example, if an updated image needs to be used or if you want to change from attended operation to non-attended operation, which are defined in separate images).

The loading process consists of two parts: loading of the system image and starting of the automated resources in the image.

Note

When you request to load a system image, the $RMEXSTR exit NCL procedure is executed before the starting process. This procedure may be customized at your site to perform any required tasks before any automated resources are started. The starting process cannot proceed if the exit sets a non-zero return code.

The LOAD command loads all resources in a system image.

Resources are started according to any relationships defined in the system image, and subject to resource availability.

From a focal point region in a multisystem environment, you can use the LOAD command to load a system image in any linked region.

When you issue a LOAD command from a stand-alone region, the LOAD Command Parameter Specification panel is displayed.

When you issue a LOAD command from a region in a multisystem environment, the Execute LOAD Command panel is displayed first. This panel lists all the connected regions. Enter **S** beside the region in which you want to load a new system image. The LOAD Command Parameter Specification panel is then displayed.

Figure 14-1 shows an example of the LOAD Command Parameter Specification panel.

Figure 14-1. LOAD Command Parameter Specification Panel

```
SOLVPROD--- Automation Services : LOAD Command Parameter Specification --------
Command ===>

.- Load Command Parameters ---------------------------------------------------.
|                                                                              |
|  ACB Name ................. SOLV1                                          |
|  Current SysName .......... SOLV                                             |
|  Current Version .......... 0001                                             |
|                                                                              |
|  SysName to be Loaded ....+ SOLV____                                         |
|  Version to be Loaded ....+ 0001                                             |
|  Global Automation Mode ... MANUAL___ (Automated or Manual)                  |
|                                                                              |
|  Last Warm Start Date ........ 01-JUN-1997                                   |
|                  Time ........ 09.15.06                                      |
|                  ACB Name .... SOLV1                                         |
|                                                                              |
|  Checkpoint Restart Status ... ACTIVE                                        |
|  Perform COLD start? ......... NO_                                           |
|                                                                              |
'------------------------------------------------------------------------------'

 F1=Help      F2=Split                                            F6=Action
                           F9=Swap                              F12=Cancel
```

Specify the system image you want to load and the global operation mode of the loaded image. (See the section, *Changing the Global Operation Mode*, on page 14-2 for information about the global operation mode.) If the checkpoint restart feature is enabled, you can specify whether you want to perform a cold load or a warm load. (See the section, *Cold and Warm Load Features*, on page 14-7 for information about these two types of loading methods.) Press F6 (Action). A Confirmation panel is displayed. Figure 14-2 is an example.

Figure 14-2. Confirmation Panel

```
SOLVPROD--------- Automation Services : Command Confirmation ------------NET001


OpSys    MVSESA                                                Userid USER01
Vtam Ver 4.3.0                                                 LU     NMMAF065
Sub Area SYDVTM01                                              THU 01-JUN-1997
                                                               Time   09.15.17

*** WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING ***
*** WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING ***
*** WARNING                                                       WARNING ***
*** WARNING    Command to Confirm .... LOAD                       WARNING ***
*** WARNING    SysName and Version ... SOLV 0001                  WARNING ***
*** WARNING    Parms : NEWSYS=SOLV NEWVERS=0001 MODE=MANUAL WARM  WARNING ***
*** WARNING                                                       WARNING ***
*** WARNING    Enter CONFIRM to continue                          WARNING ***
*** WARNING    Enter CANCEL or press PF12 to terminate            WARNING ***
*** WARNING                                                       WARNING ***
*** WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING ***
*** WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING ***

Response ... _______  ( CONFIRM / CANCEL )
 F1=Help      F2=Split
                          F9=Swap                              F12=Cancel
```

Enter **CONFIRM** in the Response field to start the loading operation. A message is displayed on your screen to indicate the acceptance of the LOAD command.

When loading starts, the Initialization Status panel is displayed to users of the status or graphical monitor in the region where the system image is being loaded. Figure 14-3 is an example.

Figure 14-3. Initialization Status Panel

```
SOLVPROD---------- Automation Services : Initialization Status ----------------
Command ===>

     . Initialization Status ----------------------------------------.
     |                                                               |
     |    System Name ............. SOLV                            |
     |    Version ................. 0001                            |
     |    Automation Global Mode ... AUTOMATED                       |
     |    Initialization Started ... 09:15:59                        |
     |    Current Time ............ 09:16:09                        |
     |                                                               |
     |    Initialization Status .... RELOADING                       |
     '---------------------------------------------------------------'
     . Information -------------------------------------------------.
     |                                                               |
     |    The selected function cannot be accessed until the         |
     |    Initialization Status is COMPLETED.                        |
     |                                                               |
     |    To refresh the above information press the Enter key.      |
     |    To view detailed status information press the Action key.  |
     |    To return to the previous panel press the Exit key.        |
     '---------------------------------------------------------------'
 F1=Help      F2=Split     F3=Exit                              F6=Action
                           F9=Swap
```

Press ENTER to monitor the status of the loading process. Messages generated during the process are logged. If the process fails, press F5 (FullLog) to view these messages for the possible cause of the failure. After you correct the error, press F6 (Load) to reload the system image. You can change the system image to be reloaded.

Swapping the Active System Image

During system operations, you might want to change the view of the system.

To implement different views of the system, you make copies of the system image and update the copies to reflect the views. You can then use the LOAD command to swap between the different images.

System image loading time depends on the complexity of the system view. To improve performance, you should ensure that command cache is available. The section, *Tuning the CPU Time Consumed by the Loading of a System Image*, on page 14-7 describes the tuning that is available to the image loading operation.

Cold and Warm Load Features

If the checkpoint restart status is INACTIVE, a system image is loaded by using the cold load method. The status of all the resources defined to the image are checked and, if required, automated resources are brought into the desired state. Previous manual overrides performed on the resources in that image are not preserved.

If the checkpoint restart status is ACTIVE, you can also use the warm load method. This method enables you to preserve the overrides across reloads of the image.

The checkpoint restart status is set in the AUTOIDS parameter group. Refer to the *Automation Services Administrator Guide* for information about parameter groups.

Tuning the CPU Time Consumed by the Loading of a System Image

Depending on the state of the system, loading of a system image can consume up to 100% of the time of a single CPU at certain times, possibly affecting the performance of other jobs. Therefore, you should load the system image during the initial program load (IPL). (In a multiprocessor system, the maximum total CPU time consumed is correspondingly less than 100%. For example, in a dual processor system, the loading operation can consume up to 50% of the total CPU time.)

However, if you need to load a system image during production hours, you can do one or more of the following to reduce the instantaneous CPU consumption:

- Allocate the active command cache in the AUTOTABLES parameter group. The cache reduces the processing required to find out the status of defined resources. This method reduces total CPU consumption.
- Reduce the number of consoles that can be acquired by the region (but maintain at least five consoles). This method reduces the instantaneous CPU consumption but increases the time for the system image to be loaded. Total CPU consumption remains unchanged.
- Place the task in a lower performance group. This method also reduces the instantaneous CPU consumption but increases the time for the system image to be loaded. Total CPU consumption remains unchanged.

Shutting Down Resources in an Active System Image

You can shut down the resources in an active system image from the status or the graphical monitor. You can shut down resources in a system image on the local or a remote region. Once a shutdown is in progress, you cannot cancel the operation. However, you can issue the STARTSYS command to restart the resources.

Note

When you request to shut down a system image, the $RMEXSHT exit NCL procedure is executed before the actual shutdown starts. This procedure may be customized at your site to perform any required pre-shutdown tasks. The shutdown process cannot proceed if the exit sets a non-zero return code.

You can monitor the shutdown process from both the status and graphical monitors. After the shutdown, you can still operate individual resources by using line commands.

There are two shutdown commands: SHUTSYS and SHUTFORCE. These commands are effective only if the global operation mode is AUTOMATED (see the section, *Changing the Global Operation Mode*, on page 14-2).

Shutting Down Automated Resources Only

The SHUTSYS command stops all resources that are in the AUTOMATED operation mode. The command overrides the desired states of all the resources in the system image with the value INACTIVE to initiate the shutdown.

To shut down resources that are in the MANUAL or IGNORED operation mode, either set the resource operation mode to AUTOMATED before issuing the SHUTSYS command or stop those resources manually. Type **MA** next to the resources for which you want to set the mode to AUTOMATED or type **T** next to the resources that you want to stop manually, and press ENTER.

Use the SHUTSYS command to specifically shut down automated resources for system maintenance purposes.

Caution

If the region is self-managed (that is, defined to the active system image) and you want the region to remain available, ensure that the region resource is in the MANUAL operation mode before you issue the SHUTSYS command.

The syntax is as follows:

```
SHUTSYS      [SYSNAME=systemname] [VERSION=nnnn]
```

SYSNAME=	Specifies the name of the system image you want to shut down.
VERSION=	Specifies the version of the system image you want to shut down. You must specify a 4-digit value (for example, 0001).

You can issue the SHUTSYS command with or without parameters from the monitor. If you do not specify parameters, the Execute SHUTSYS Command panel is displayed. This panel lists the active system images in all connected regions. Enter **S** beside the system image you want to shut down.

A confirmation panel is displayed. If you wish to cancel the operation, press F12 (Cancel) to cancel the shutdown. Otherwise, enter **CONFIRM** to shut down the automated resources in the system image.

Shutting Down All Resources

The SHUTFORCE command sets all resources that are in the MANUAL or IGNORED operation mode in a system image to the AUTOMATED operation mode, then shuts down all the resources that are in the AUTOMATED operation mode by overriding their desired states with the value INACTIVE. The command is effective only if the global operation mode is AUTOMATED.

Caution

If the region is self-managed, issuing the SHUTFORCE command shuts down the region itself.

The syntax is as follows:

```
SHUTFORCE   [SYSNAME=systemname] [VERSION=nnnn]
```

SYSNAME=	Specifies the name of the system image you want to shut down.
VERSION=	Specifies the version of the system image you want to shut down. You must specify a 4-digit value (for example, 0001).

You can issue the SHUTFORCE command with or without parameters from the monitor. If you do not specify parameters, the Execute SHUTFORCE Command panel is displayed. This panel lists the active system images in all connected regions. Enter **S** beside the system image you want to shut down.

A confirmation panel is displayed. If you wish to cancel the operation, press F12 (Cancel) to cancel the shutdown. Otherwise, enter **CONFIRM** to shut down the automated resources in the system image.

Restarting Resources in an Active System Image

If the region is still available after a SHUTSYS or SHUTFORCE command, you can use the STARTSYS command to restart the resources that have been shut down.

Note

When you request to restart a system image, the $RMEXSTR exit NCL procedure is executed before the actual restart. This procedure may be customized at your site to perform any required tasks before the starting process. The starting process cannot proceed if the exit sets a non-zero return code.

The syntax of the STARTSYS command is as follows:

```
STARTSYS      [SYSNAME=systemname] [VERSION=nnnn]
```

SYSNAME=	Specifies the name of the system image you want to restart.
VERSION=	Specifies the version of the system image you want to restart. You must specify a 4-digit value (for example, 0001).

You can issue the STARTSYS command with or without parameters from the monitor. If you do not specify parameters, the Execute STARTSYS Command panel is displayed. This panel lists the active system images in all connected regions. Enter **S** beside the system image you want to restart.

A confirmation panel is displayed. If you wish to cancel the operation, press F12 (Cancel) to cancel the restart. Otherwise, enter **CONFIRM** to restart the resources in the system image. The restart process removes the INACTIVE desired state overrides from the resources to initiate the startup.

Effect of Checkpoint Restart on the STARTSYS Command

The STARTSYS command behaves differently depending on the following:

- Whether the command is issued after a SHUTSYS or a SHUTFORCE command.
- Whether checkpoint restart is enabled in the AUTOIDS parameter group. To access the list of parameter groups, enter the **/ICS** shortcut.

Behavior of the STARTSYS Command After a SHUTSYS Command

After a SHUTSYS command, the STARTSYS command preserves previous manual overrides only if checkpoint restart is enabled.

Behavior of the STARTSYS Command After a SHUTFORCE Command

The SHUTFORCE command forces all resources into the AUTOMATED operation mode. Resources that are originally in the MANUAL mode are also shut down. When you restart the resources after a SHUTFORCE command, the following occurs:

If checkpoint restart is …	Then previous manual overrides are …
Disabled	Not preserved and all resources are automated.
Enabled	Preserved and resources that were in the MANUAL mode are not restarted.

Part III

Event Management

15

About Event Management

This chapter introduces the EventView feature of Automation Services.

Note

EventView is part of the automation feature. Some products do not use the automation feature. These products use Automation Services to provide only knowledge base management and presentation services.

This chapter contains the following topics:

- What is EventView?
- Benefits of Using EventView
- Developing Event Management Rules
- Selecting the Messages to Be Monitored
- Generating Alerts

What is EventView?

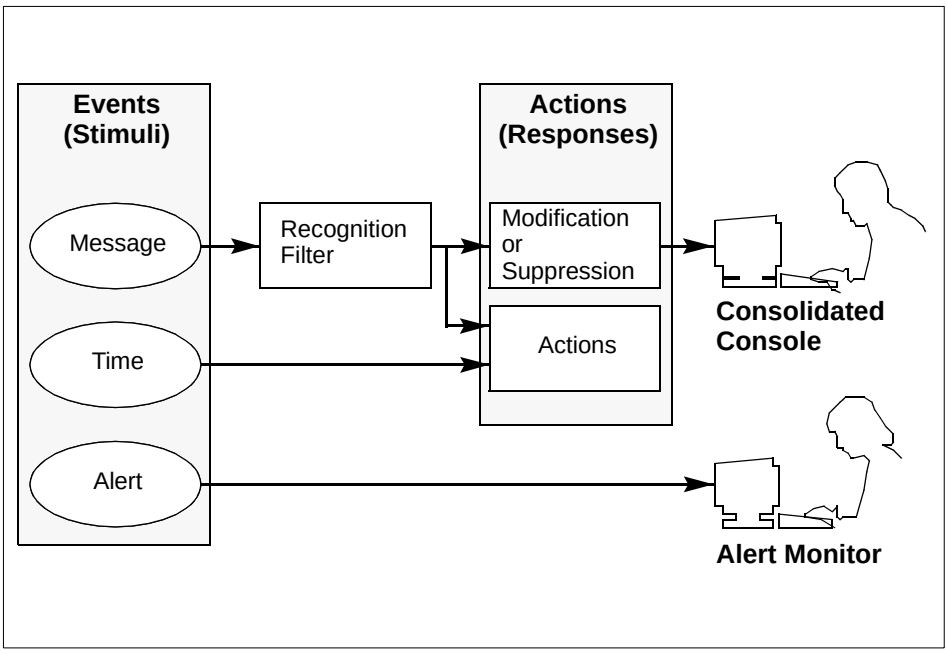
EventView performs automation at the event level (see Figure 2-1 on page 2-3). It provides event level automation and control, and can handle timed events.

The event level is where messages are generated. Successful event management relies on the recognition of significant events from amongst the mass of messages and alerts being transmitted through a system, and appropriate responses to these events.

EventView provides the following functions:

- Event-based automation, which relies on the following:
  - The creation of appropriate rules and rule sets
  - The processing of messages
  - The processing of EventView timers
  - Message learning
- Console message consolidation
- Alert monitoring

Figure 15-1. EventView Processing



Event-based Automation

You can define event rules to do the following:

- Suppress messages
- Change message text
- Enhance message presentation (for example, highlighting)
- Set route and descriptor codes
- Perform actions

Rules are grouped logically into rule sets, which define how an event will be processed and what actions will be taken in response to an event that is *not* related to a resource. (Resource-based events are handled by ResourceView.) An event can be either a message or a specified time.

Sample Message Suppression Rule Sets

EventView provides the following samples of message suppression rule sets:

- AGRSUPP, which is based on the aggressive list of suppressible messages recommended by IBM
- CONSUPP, which is based on the conservative list of suppressible messages recommended by IBM

See the MVS initialization and tuning reference manual for those lists.

The corresponding rule sets for MSP systems are AGRSPMSP and CONSPMSP. The corresponding rule sets for VOS3 systems are AGRSPVOS and CONSPVOS.

Console Message Consolidation

You can monitor message flows from multiple consoles on a single screen—the consolidated console. Console consolidation controls the way you see messages on the console. In addition, messages displayed on the consolidated console are affected by EventView processing.

For example, message text and message presentation can be modified by EventView, and the consolidated console user sees the modified message. If EventView suppresses a message, that message is not displayed on the consolidated console.

You can define message profiles that customize the view of the message flow. Different users can have different sets of message profiles to suit the functions they perform. See the *Automation Services Administrator Guide* for details of how to specify the message profiles for a user.

Message profiles enable the meaningful grouping of messages based on criteria such as system, message ID, job name, and system codes.

Alert Monitoring

The alert monitor displays alerts that warn you of problems or impending operational problems. These problems might or might not relate to resources and services. For alerts related to resources or services, you can use the status or graphical monitors to obtain a different view of the problem.

Alerts can also be used to indicate significant events (for example, reminding you to perform a task that cannot be automated).

When you receive an alert, you can perform the following:

- Select the alert to display any recommended actions.
- Access the status monitor or the transient log if the alert is related to a service or resource.
- Raise a problem ticket for the alert.
- Enter notes about the alert.
- Access the alert history log.

You can use the GENALERT and DELALERT macros to generate alerts and delete alerts.

Benefits of Using EventView

EventView benefits your organization in the following ways:

- Reduces system console message rates; you can filter messages received and suppress unwanted messages
- Produces a standardized response to events or problems
- Enables you to schedule actions to occur at specific times or at regular intervals
- Gathers useful statistics for messages and timers
- Able to learn messages
- Enables you to monitor message flows to multiple consoles on a single screen
- Enables you to generate alerts to remind operators of significant events

Developing Event Management Rules

Rule sets consist of a number of rules that define how an event is to be processed and what actions are to be taken in response to the event.

Use a number of rule sets to organize your rules in logical and manageable groups.

For a rule set to be in an active state, it must be associated with an active system image. However, only one rule set can be associated with an image. If you want more than one rule set to be active, you need to include all the other required rule sets in the rule set that is associated with the active image.

Typical stages in implementing a rule set are discussed below and shown in Figure 15-2.

Step 1. Identify the rule sets you want to create.

Step 2. Create the primary rule set that is to be associated with the active image.

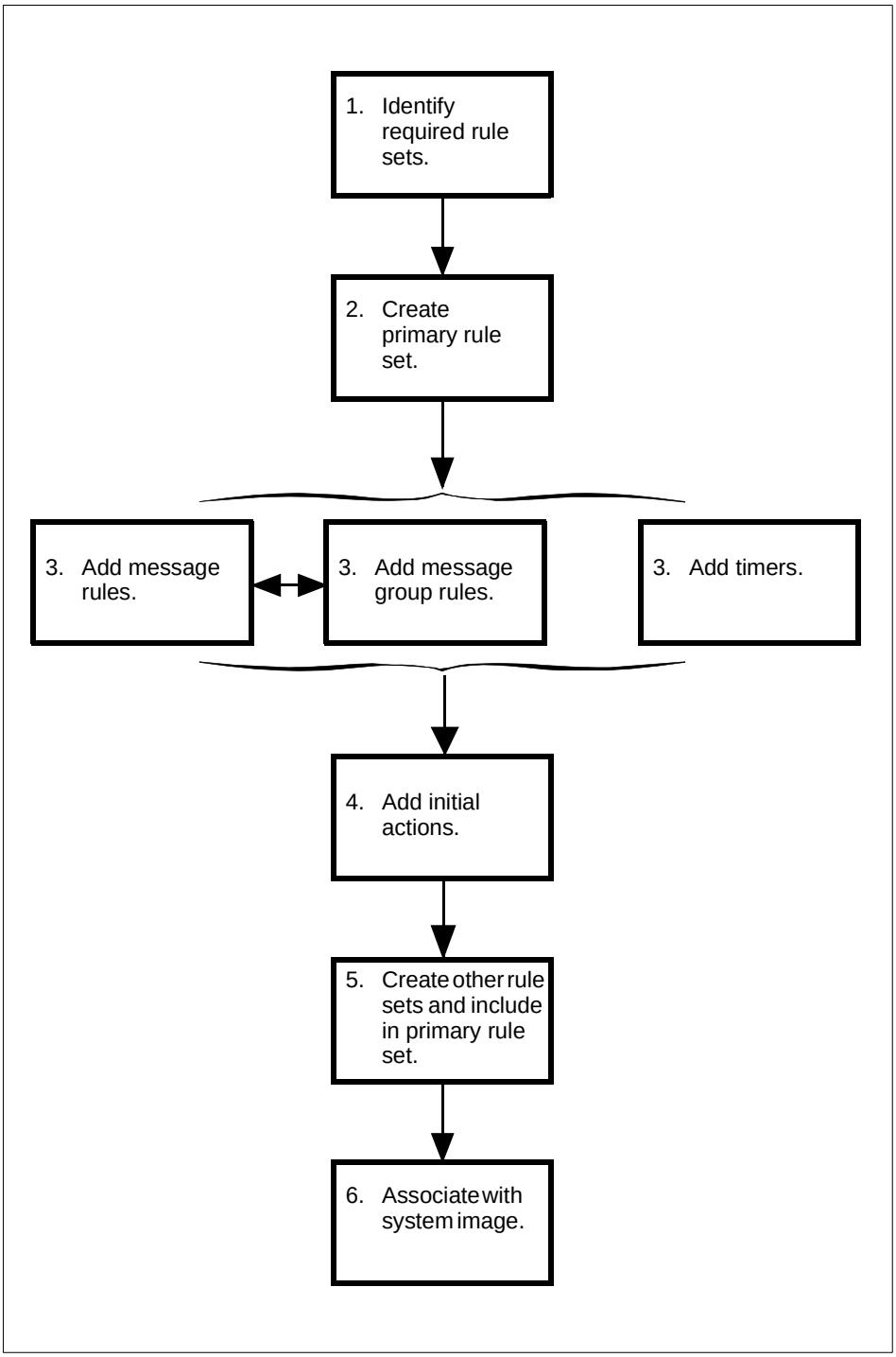
Step 3. Add message rules, message group rules, and timers.

Step 4. Add actions that need to be performed before normal rule processing occurs.

Step 5. Create the other rule sets, and include them in the primary rule set.

Step 6. Associate the rule set with the appropriate image.

Figure 15-2. Implementing a Rule Set



Chapter 16, *Defining and Maintaining Rulesets*, describes how to add rule sets and associated rules, and how to include rule sets in other rule sets.

Selecting the Messages to Be Monitored

Besides responding to resource status, you need to respond to events that are not handled by resource automation.

The Automation Services components that affect message display are the EventView message rules and the console message consolidation facility. To use the latter facility, you need to define message profiles.

Normally, you use both components and you can monitor messages from multiple systems. However, if you do not want to define message profiles, you can disable the message consolidation facility and only messages from the local system can be monitored. You control the availability of the facility by using the CCONSOLIDATN parameter group. (See the *Automation Services Administrator Guide* for information about parameter groups.)

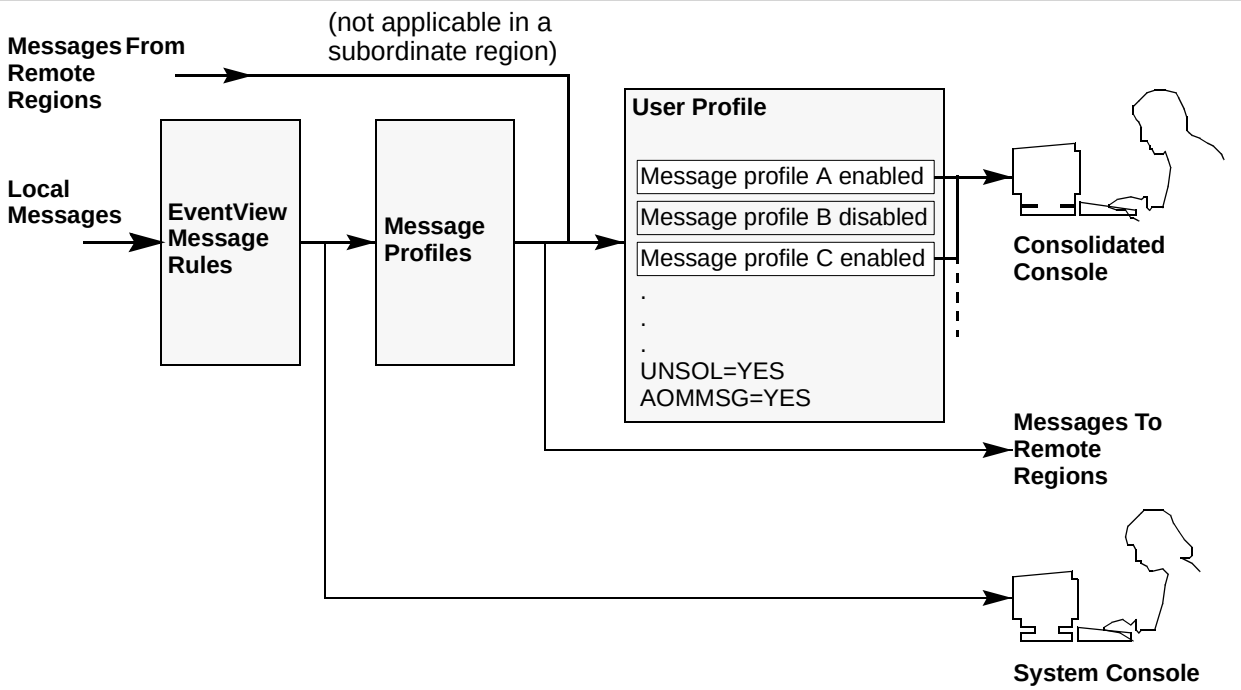
Message Consolidation Enabled

With message consolidation enabled, the message monitor becomes a consolidated console. Using the console message consolidation facility, an operator is able to monitor messages from one or more systems on the consolidated console.

You use EventView message rules to preprocess the messages (for example, suppressing or highlighting the messages). You can then use message profiles to select the type of processed information to display on the consolidated console. For example, you can define a message profile that selects messages from a particular system. You can selectively enable profiles to customize their view of monitored events (for example, CICS messages only).

Figure 15-3 shows how messages arrive at the consolidated console.

Figure 15-3. Console Message Consolidation



For information about how to define EventView message rules, see Chapter 16 and Chapter 17.

For information about how to define message profiles, see Chapter 20.

For information about how to define user profiles, see the *Automation Services Administrator Guide*.

Typical stages in implementing message profiles are discussed below and shown in Figure 15-4.

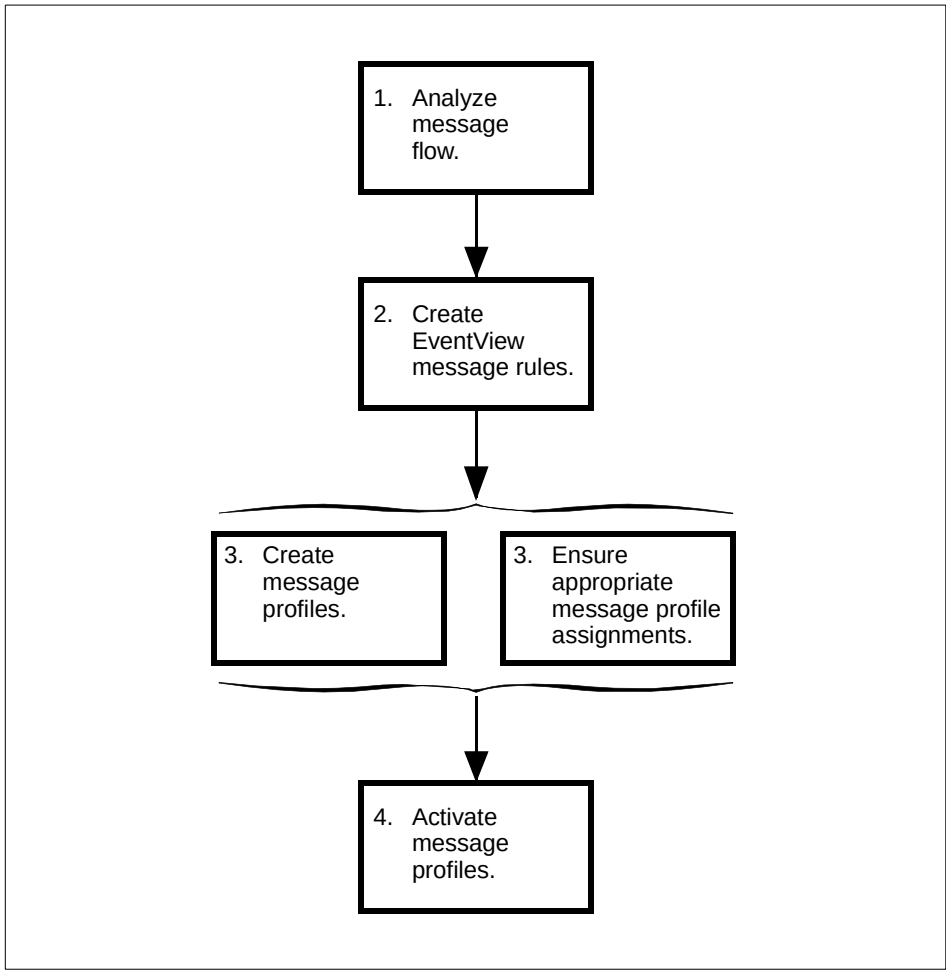
Step 1. Analyze the message flow and the operations tasks to determine the different message views that are required.

Step 2. Create EventView rules to suppress unwanted messages.

Step 3. Create the message profiles, and ensure that each operator is assigned the appropriate message profile IDs in the user definition and user profile.

Step 4. Activate the message profiles.

Figure 15-4. Implementing Message Profiles



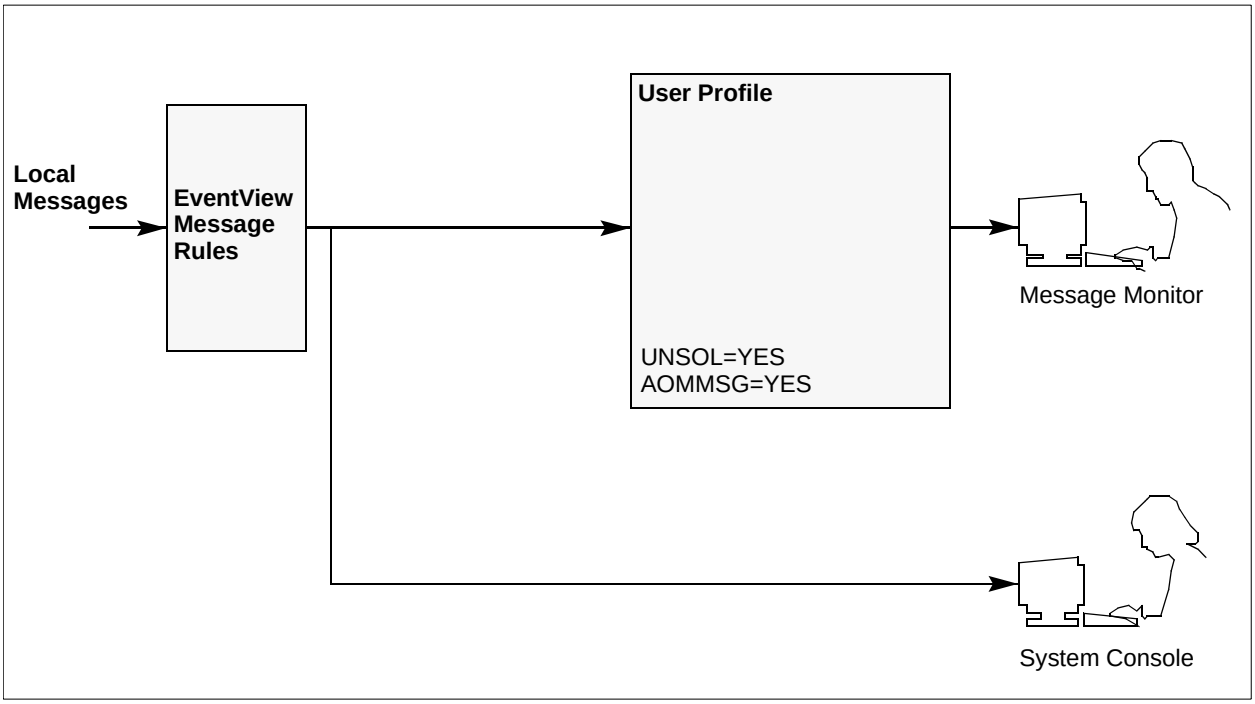
Chapter 20, *Defining and Maintaining Message Profiles for the Consolidated Console*, describes how to create and activate message profiles.

Message Consolidation Disabled

Without the console message consolidation facility, you are able to monitor local messages only. Remote messages are not routed to this region, and messages from this system are not routed to remote regions.

Figure 15-5 shows how messages arrive at the message monitor.

Figure 15-5. Console Message Consolidation Disabled



Generating Alerts

Alerts are displayed on the alert monitor. Alerts can be internally generated to warn you of problems or impending problems in systems operations. Alerts can also be user-generated. Generate alerts through user-defined processes by using the following macros:

- GENALERT enables a process to generate an alert of a specified severity.
- DELALERT enables a process to remove an alert from the alert monitor.

Use alerts to warn operators of significant events (for example, reminding the operator to perform tasks that cannot be automated).

For information about how to define processes, see Chapter 8, *Defining Processes to Implement Complex Operations*.

16

Defining and Maintaining Rulesets

This chapter describes how to create rulesets.

This chapter contains the following topics:

- Working With Rulesets
- Adding Associated Rules
- Adding Initial Actions
- Including Rule Sets in Other Rule Sets
- Maintaining Rule Sets
- Using EventView Variables

Working With Rulesets

Rulesets consist of various members that define how an event is processed and what actions are taken in response to the event. A ruleset might include:

- Initial actions
- Message rules
- Message group rules
- Timer rules
- Other rulesets

You might want to create a ruleset for each area of responsibility. For example, you might create a CICS ruleset, a VTAM ruleset, and so on, to organize your rules in logical and manageable groups.

Note

Automation that deals with the status of resources should be specified in the resource definition, *not* in a message rule.

For a rule set to become activated, it must be associated with an active system image. System images are described in Chapter 4, *Working With System Image Definitions*.

Only one rule set, known as the primary rule set, is associated with a system image. Therefore, if you want more than one rule set to be activated, you need to include all the other required rule sets in the rule set that is associated with the active image. You could, for example, create a master rule set into which all other rule sets are included. See the section, *Including Rule Sets in Other Rule Sets*, on page 16-11 for details of how to include a rule set in another rule set.

Adding a Rule Set

You need to add a rule set before you can add the associated members. To add a rule set, you do the following:

Step 1. Enter the **/EADMIN.R.R** path on the primary menu, to display the Ruleset List panel.

Step 2. Use F4 (Add) to add a rule set.

Step 3. Complete the resulting Ruleset Description panel, shown in Figure 16-1, adding comments on the Comments panel if required. Refer to the online help for assistance.

Note

A rule set can be activated only if it has an ACTIVE status.

Figure 16-1. Adding a Rule Set

```
SOLVPROD--------------- EventView : Ruleset Description -----------------------
Command ===                                                     Function=ADD

Ruleset Name ........ ________                    Ruleset Status ...+ ACTIVE__
Short Description ... ______________________________

. Control Options -------------------------------------------------------------.
|                                                                              |
| Default Message Delivery ....+ YES  (if no rule for message)                 |
| Perform Message Modification?  YES                                           |
| Perform Action? ............. YES                                           |
| Log Ruleset Activity? ........ NO_                                           |
|                                                                              |
| Collect Statistics? .......... YES                                           |
| Learn New Messages? .......... YES                                           |
|                                                                              |
'------------------------------------------------------------------------------'
. Notes -----------------------------------------------------------------------.
| The Control Options in this Ruleset will not be effective as the Control    |
| Options in Ruleset XXXXXXXX activated by System Image YYYYYY Version 0001   |
| take precedence.                                                             |
'------------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=File      F4=Save
              F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Specifying Control Options

The control options for the primary rule set override those specified for included rule sets.

When setting up a rule set, you might want to test it without actually triggering any rules. To do this, you would set the Perform Message Modification? and Perform Action? flags to NO, and the Log Ruleset Activity? flag to YES. You could then see from the entries in the general activity log what activity would take place if the rule set was, in reality, working as intended.

Monitoring Rule Set Status

The status of the active rule set and all its included rule sets on the current system can be viewed by entering the **/EADMIN.S.R** path to display the EventView : Ruleset Status panel, which is shown in Figure 16-2.

Figure 16-2. Monitoring Rule Set Status

```
SOLVPROD---------------- EventView : Ruleset Status -------------------------
Command ===>                                                  Scroll ===> PAGE

Primary Ruleset ............... BACKUP
Short Description ............. Backup task ruleset.
Default Message Delivery ....... YES
Perform Message Modification ... YES
Perform Action ................ YES
Log Ruleset Activity .......... NO
Collect Statistics ............ YES
Learn New Messages ............ YES

 Loaded Rulesets                                                    Status
********************************* TOP OF DATA ********************************
 BACKUP    Backup task ruleset.                                     ACTIVE
  INC       Included ruleset                                        ACTIVE
****************************** BOTTOM OF DATA ******************************




 F1=Help      F2=Split     F3=Exit      F4=Return    F5=Find      F6=Refresh
 F7=Backward  F8=Forward   F9=Swap     F10=Left     F11=Right
```

This panel displays the same information as the Ruleset Description panel, plus it lists loaded rule sets. The primary rule set is the first rule set listed, followed by its included rule sets. Each level of further inclusion is indicated by indentation.

Note

If a rule set has a status of inactive, its included rule sets are not processed.

Gathering Statistics

If you specify Yes in the Collect Statistics? field on the Ruleset Description panel, then EventView collects statistics relating to messages received and timer schedule items executed. You can use these statistics to measure the effectiveness of your EventView rules.

If the SMFDATA region parameters are configured, the statistics are output to SMF (or SMS on VOS3 systems) at a user-defined interval. The SMF record format is shown in the *Automation Services Administrator Guide*.

Changing the Rule Set Associated With a System Image

You can change the rule set associated with a system image by updating the EventView Ruleset to Activate field on the System Image Definition panel. To get to the System Image Definition panel for the system image you want to update:

Step 1. Enter the **/RADMIN.I** path.

Step 2. Apply the **U** (Update) action to the listed system image that you want to update.

Step 3. Enter the new rule set name in the EventView Ruleset to Activate field, or select a rule set from the prompted field value list.

Step 4. Save the updated record.

Note

By default, EventView rule actions are not executed if the system image is operating in the MANUAL global operation mode. The actions, however, can be enabled by using the Perform Action in Manual Mode? field of the AUTOIDS region parameter group.

Adding Associated Rules

Once you have created a rule set, you can add associated message, message group, or timer rules. From the Ruleset List panel, you proceed as follows to add a rule:

Step 1. Apply the appropriate action, such as **M** (Message List), to the rule set with which you want to associate the new rule.

Step 2. Press F4 (Add).

Step 3. Complete the fields on the initial panel displayed, and on any subsequent panels as required.

Adding Message Rules

Message rules contain some or all of the following information:

- Message text and filtering criteria
- Message delivery and suppression details
- Required message modification details
- Which actions are triggered by a message
- Which message groups the current message rule is related to
- User-defined EventView variables

Message rules are added in the same way as other rule set members—see the section, *Adding Associated Rules*, on page 16-5 for the procedure. You apply the **M** (Message List) action to the rule set with which you want to associate the new message rule.

Caution

Message Text is a mandatory field. If you enter the wildcard character in this field, *all* messages will be tested against this rule.

See Chapter 17, *Processing Messages*, for assistance in completing the fields on the message rule panels.

Message Execution Conditions

You can specify execution conditions in this panel window. The rule will only execute if all the given conditions apply. The section, *Specifying Execution Conditions*, on page 17-6 describes how to set these conditions.

Adding Timers

If you want a rule triggered on a particular day of the week (or year) and at a particular time, you need to add a timer rule. Timer rules contain the following information:

- Whether the timer rule applies to a specific system
- Up to 99 detailed schedule items
- Which actions are to be triggered by a timer
- User-defined EventView variables

Timer rules are added in the same way as other rule set members—see the section, *Adding Associated Rules*, on page 16-5 for the procedure. You apply the **T** (Timer) action to the rule set with which you want to associate the new timer.

Timers are discussed in depth in Chapter 18, *Defining Timers*.

Grouping Messages

If a message on its own is not significant, but the occurrence of another message increases its significance, then you need to create a message group to associate these messages.

Note

The order in which the grouped messages occur is not important, as long as all arrive within the specified time interval.

Adding Message Group Rules

Message group rules are added in the same way as message rules, except that you apply the **G** (Group) action to the item on the Ruleset List with which you want to associate the new group. Message group rules contain the following information:

- The maximum time interval within which all messages in the group must be received, in order to trigger the rule
- Message text for up to ten messages, on the Message Group Details panel (displayed automatically when a message is associated with a message group—see the following section, *Associating Message Rules With Message Groups*)
- The text of a message that is to be issued if the group rule is triggered, and where and how to display this message
- The action or actions to be performed when a group rule is triggered
- User-defined EventView variables, which you can flag to be set to the specified values before or after other rule actions

Associating Message Rules With Message Group Rules

To establish a relationship between a message rule and a message group rule, you need to add an entry on the Related Message Group panel (the fifth panel in the sequence of Message Rule panels, shown in Figure 16-3). You also need to add the message rule itself on the Message Group Details panel of the message group rule definition. The same message rule can be associated with up to five message group rules.

Figure 16-3. Associating Messages and Message Groups

```
SOLVPROD---------------- EventView : Message Filter --------------------TAPEMON
Command ===> 5                                                Function=UPDATE

Ruleset Name ........ TAPEMON                        Rule Status ...+ ACTIVE__
Short Description ... Mount request processing______

. Expected Message ---------------------------------------------------------.
|                                        S=ListPanels EV=ExtFilter TV=TestVars |
|      Message Text  ( WildChar = * )                                  ExtFlt |
| ___  IEC501A                                                         NO     |
'-----------------------------------------------------------------------------'
```

```
SOLVPROD---------- EventView : IEC501A Related Message Group -----------TAPEMON
Command ===>                                                  Function=UPDATE

. Message Group Table ------------------------------------------------------.
|                                                                            |
| MsgGroupID   CorrelationKey                                                |
| GROUP1______ &ZMSGJOBNM___________________________________________________ |
```

```
SOLVPROD-------------- EventView : Message Group Details ---------------TAPEMON
Command ===>                                                  Function=UPDATE

Ruleset Name ......... TAPEMON
Message Group Name ... GROUP1                        Rule Status ...+ ACTIVE__
Short Description .... Tape mount group______________
Interval ............ 00.10.00

. Expected Message ---------------------------------------------------------.
|                                                     S/B=Browse U=Update |
|      Message Rule Text                                                     |
| ___ IEC501A                                                                |
| ___ IEC509A                                                                |
| ___                                                                        |
```

On the Related Message Group panel, you need to identify the message group rule and can optionally specify a correlation key for precise recognition purposes. The correlation key enables one message group rule to cover numerous different situations, saving you from having to create numerous different rules. The rule is not triggered unless the values of the correlation keys in each of the grouped messages match.

For example, in Figure 16-3 on page 16-8, the correlation key is the name of the variable that contains the job name. The group rule would only be triggered if the messages associated with the group:

- All arrived within the specified interval (10 minutes)
- Were all generated by the same job

Note

A message group rule with only one associated message is ignored.

Adding Initial Actions

Initial actions are actions performed when a rule set is activated (that is, when the associated system image becomes active), and before message processing commences.

You add initial actions from the Ruleset List by applying the **IA** (Initial Actions) action to the nominated rule set.

Variables that are essential to the functioning of a rule set should be set in the initial action rules.

For example, in Figure 16-4, you want to log a message to indicate that a rule set is activated.

Figure 16-4. Specifying Initial Actions

```
SOLVPROD---------------- EventView : Initial Action --------------------BACKUP
Command ===> forward                                          Function=UPDATE

Ruleset Name ........ BACKUP
Initial Action Name   NOTIFY                        Rule Status ...+ ACTIVE
Short Description ... Log a startup message

System Command ... ____________________________________________________
                   ____________________________________________________
MS Command .......... LOG RULESET BACKUP IS NOW ACTIVE
```

Note

If a rule set has associated included rule sets, the initial actions specified for those rule sets are also performed when the primary rule set becomes active.

If you need to set any EventView variables before or after any of the initial actions are performed (to pass parameter values, for example), press F8 (Forward) to go to the Set Variables panel.

On the Set Variables panel, shown in Figure 16-5, you supply a name for each EventView variable that you want to set, plus the required variable value. Note that, when you use the variable subsequently, you prefix the name with &ZREV, which the EventView variable identifier.

Figure 16-5. Setting EventView Variables

```
SOLVPROD---------------- EventView : Set Variables ---------------------BACKUP
Command ===>                                                   Function=UPDATE

Set Variables Before Rule Execution? ... NO_


. Set EventView Variables ---------------------------------------------------.
|                                                                            |
|      Name       Value                                                      |
| ZREV STARTUP_ = XYZ                                                        |
| ZREV _________ = ___________________________________________________________ |
| ZREV _________ = ___________________________________________________________ |
```

Execution of Initial Actions

When a rule set is activated, the associated initial actions are executed. When you load a system image that contains a rule set that is already active, the region does not reactivate that rule set and the associated initial actions are not executed (for example, when you switch images that use the same rule set).

If you have several system images that use the same rule set and you want the initial actions associated with the rule set executed every time you load one of those images, you can create a primary rule set for each of the images. Each primary rule set includes the actual rule set you want. Because the primary rule sets are different, it is activated every time you switch between the images, thus executing the initial actions.

Including Rule Sets in Other Rule Sets

You can include a rule set in another rule set by doing the following from the Ruleset List panel:

Step 1. Apply the **I** (Include) action to the rule set in which you want to include another rule set.

The Included Ruleset List panel is displayed. This list is blank if there are no rule sets included in the current rule set.

Step 2. Press F4 (Add) to include a rule set.

Step 3. From the Eligible Ruleset List panel, select the rule set to be included in the current rule set.

The selected rule set is added to the Included Ruleset List for the current rule set. This means that the included rule set is active when the parent rule set is active.

Note

If you update an included rule set, only the control options of the rule set invoked by the system image are used.

Maintaining Rule Sets

You can browse, update, copy, and delete rule set definitions from the Ruleset List panel.

The C and the D action codes enable you to copy and to delete an *entire* rule set. To copy or delete the rule set definition only, use the CO or DO action codes. You can use the DO action code to delete a rule set only if it is empty—that is, it contains no rules.

Using EventView Variables

The ability to set and use EventView variables within rules enables you to create dynamic rules that depend on conditions identified by other rules and rule sets. That is, you use EventView variables to control rule execution.

EventView variables can be used:

- To pass information and data between rules
- To obtain more information about the environment in which the rule is executing
- To record system states

EventView variables can be set on the Set Variables panel of a message rule, a timer rule, or a group rule. Here, you can set values for up to six variables. These values can be literal, or you can specify a substitution variable as the source of the variable value for a message rule. See Appendix D, *Variables*, for a list of variables.

EventView variables can be used by:

- *Rules*, to do the following:
  - Provide a correlation key value to be matched, on the Message Delivery panel and the Related Message Group panel.
  - Provide a value for insertion in replacement text. Replacement text specified on the Message Modification, Set Variables and Test Variables panels can include EventView variable names.
- *Processes*, where the macros EVVARGET and EVVARSET can be used to get and set the values of EventView variables. Variable names can be specified in the Parameters field on the Rule Action panel, as well as on other panels where processes are invoked. See also Chapter 8, *Defining Processes to Implement Complex Operations*, and Appendix E, *Macros*.
- *NCL procedures*, where the $RECALL application program interface (API) can be used to get and set the values of EventView variables. The *Automation Services Administrator Guide* describes the $RECALL API.

You must remember to add the EventView variable indicator prefix, &ZREV, to a variable name when it is specified for evaluation.

Viewing EventView Variables

You can view all EventView variables that have been set by entering the **/EADMIN.S.V** path to display the EventView : Active Variables panel.

17

Processing Messages

This chapter describes how to use EventView message rules to process messages.

This chapter contains the following topics:

- Specifying Message Filtering Criteria
- Specifying Message Delivery
- Modifying Messages
- Specifying Actions to be Taken in Response to Messages
- Suppressing Messages

Specifying Message Filtering Criteria

When a message is received, the message text is compared with the scan text specified on the Message Filter panel, which is a primary key to message recognition. For example, if you specify TESTMSG1 as the scan text, any message starting with those eight characters is considered to be a match, including TESTMSG12, and TESTMSG1 TESTING.

Note

If you want to capture a message that has leading blanks, you do not need to specify the leading blanks on the message filter panel. However, on the Extended Message Filter panel, absolute position is important so leading blanks must be counted when using start position of text.

This message text can include wildcard characters. The default is the asterisk (*). You can simply specify the message text that will trigger the rule if the execution conditions are met. You can also specify additional filters on further panels, by typing **E** beside the message text (as shown in Figure 17-1), to check for a variety of different conditions.

Figure 17-1. Specifying Extended Filtering Criteria

```
SOLVPROD---------------- EventView : Message Filter --------------------TAPEMON
Command ===>                                                   Function=UPDATE

Ruleset Name ........ TAPEMON                        Rule Status ...+ ACTIVE
Short Description ... Mount request processing

. Expected Message ------------------------------------------------------------.
|                                      S=ListPanels E=ExtFilter T=TestVars |
|      Message Text  ( WildChar = * )                                  ExtFlt |
| e__  IEC501A                                                         NO     |
```

```
SOLVPROD--------------- EventView : Extended Message Filter -------------------
Command ===>                                                   Function=UPDATE

Message Text ......... IEC501A
Wildcard Character ... _
Descriptor Code .....+ _____________
Route Code ..........+ _____________
Message ID ........... _____________ (of major line)
System Name .......... ________

. Message Text Analysis -------------------------------------------------------.
|   Strt Word      Scan                                                       |
|   Pos  Num  Opr  Text                                                       |
| 1 ___  ___  ___  ___________________________________________________________ |
| 2 ___  ___  ___  ___________________________________________________________ |
| 3 ___  ___  ___  ___________________________________________________________ |
| 4 ___  ___  ___  ___________________________________________________________ |
| 5 ___  ___  ___  ___________________________________________________________ |
|                                                                             |
| Expression ..... ____________________________________ e.g. (1 and (2 or 3)) |
'-----------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=OK
              F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Using Wildcards in Message Text

Normally, you can just specify enough message text to identify the messages you want the message rule to process.

Use wildcard characters to insert character patterns within the message text. However, if you use a wildcard character, you must also add a wildcard character to the end of the message text if required.

The following examples show the correct use of wildcard characters:

```
*EC501A*
IEC50*A*
IEC5**A*
```

The number of characters represented by a wildcard character is dependent on its position within the message text as follows:

- If the wildcard character is at the beginning of, or embedded within, the message text, it represents one character.
- If the wildcard character is at the end of the message text, it represents any number of characters.

Specifying Extended Filtering Criteria

The Extended Message Filter panel enables you to specify precise criteria to be matched, as described in the following subsections.

Wildcard Character

You can specify a value other than the default value of an asterisk (*) in the Wildcard Character field. This change is reflected in the Wildcard Character field on the Message Filter panel when you save the extended filtering criteria. This is useful if the message actually contains an asterisk.

Descriptor Code

This code determines the color that the operating system will use to display the message on a color console, and whether the message is to be a non-roll delete message. If one or more values are specified in this field, the descriptor codes assigned to a message are tested against the specified values. A message matches if it contains any of the listed descriptor codes. Appendix K, *Message Descriptor Codes*, describes the descriptor codes.

Route Code

This routing code is used by the operating system to control message delivery. If one or more values are specified in this field, the routing codes assigned to a message are tested against the specified values. A message matches if it contains any of the listed routing codes. Appendix J, *Message Routing Codes*, describes the routing codes.

Message ID

The message ID is the first word of the message text (disregarding any flag characters, such as an asterisk, in position 1 or 2). When a secondary line of a multiline WTO message is being filtered, the message ID for the line is the same as that for the primary line of the WTO message.

System Name

The name of the system from which the message originated. This is useful if the local system reissues messages received from other systems. Messages are reissued if the system is part of a sysplex environment, or is a JES3 or JSS4 global processor.

For VOS3 systems using JSS4, this criterion matches the processor ID.

Specifying Message Text Analysis Criteria

You can analyze the text of the current message by word, phrase, or string, by specifying any combination of start position, word number, and permitted operator (such as equals, is greater than, and so on). You can specify up to five tests to be performed on the message text, and even link these tests in an expression.

Note

EventView comparisons are *text-based*, that is, they are performed character by character, starting from the leftmost character of the extracted text. Text checking is done using the EBCDIC codes. Numbers are regarded as text. For example, the character string 100 is less than 99.

Text Analysis and how to complete the fields in this window are discussed in detail in Appendix F, *Message Text Analysis*.

Linking Tests in an Expression

The tests you specify in the Text Analysis box can be linked in a defined relationship in the Expression field, using the Boolean operators AND, OR, and NOT.

For example, if you specify `1 and (2 or 3)` in the Expression field, this indicates that test one must always be true, and either test two or test three must be true, before the rule can be triggered.

Note

If you leave the Expression field blank, all specified conditions must be true.

Testing EventView Variables

If there are any EventView variables specified on the Test EventView Variables panel (the second panel in the extended filter sequence), these values are compared with the value of the predefined EventView variable when the rule is validated. For the rule to be triggered, they must match.

Specifying Execution Conditions

If the message text passes the filtering process, further validation is performed to see whether the message received meets the specified execution conditions.

All the execution conditions specified on the Message Filter panel must be met before the message rule can be triggered. Figure 17-2 shows an example.

Figure 17-2. Specifying Execution Conditions

```
. Execution Conditions -------------------------------------------------------.
|                                                                              |
| Job Name ....            Rule Priority ...........      (1 is best)         |
| Job Type ....            Execute If Not Best Fit?                           |
|                                                                              |
|         Mon Tue Wed Thu Fri Sat Sun       Time       Start       End        |
| On Days NO  NO  YES NO  NO  NO  NO        Range1 ...                        |
|                                           Range2 ...                        |
|                                                                              |
'------------------------------------------------------------------------------'
```

Caution

If you want to detect a message from a started task that runs under the master scheduler (that is, by using the SUB=MSTR operand), do not use the Job Name and Job Type fields.

Dealing With Overlapping Rules

You need to take into consideration that there may be more than one rule that applies to the same message.

EventView selects and executes the rule considered to be *the best fit*. This decision is based on how specific the filtering and execution conditions are; the more specific the rule (for example, the more message text specified), the better the fit.

You can override this determination of the best fit by entering a value (in the range 1 to 99) in the Rule Priority field, to indicate the order of importance. Top priority rules are given a ranking of 1, while the least important rule can be ranked 99.

You might want to trigger multiple rules for one message. The Execute if Not Best Fit? field, which can be set to Yes or to No, functions as follows:

- If set to NO (the default), the rule is not executed unless it is the best fit.
- If set to YES, the rule actions will be executed whenever validation is successful.

Note

Only the rule that is the best fit will alter message delivery, modify the message, and reply to the message.

Specifying Message Delivery

When a message satisfies the filtering criteria of a message rule and provided that the rule is the best fit, the rule controls how the message is to be delivered. Specify the delivery criteria on the Message Delivery panel.

Setting the Deliver Flag

You set the Deliver flag to:

- YES (the default), if you want the message to be delivered to both the operating system and the consolidated console, and to be logged to the system log (SYSLOG) and the Automation Services activity log
- IGN, if you want the message to be ignored by the region, but delivered to the operating system and to be logged to the system log (SYSLOG)
- LOG, if you want the message logged to SYSLOG and the activity log, but not displayed on the console
- NO, if you want the message suppressed everywhere with the exception of SYSLOG
- Z, if you want the message suppressed everywhere, including SYSLOG

Note

Delivery of system messages to the activity log might be suppressed by the LOGFILES parameter group (see the *Automation Services Administrator Guide*).

Setting Delivery Thresholds

Thresholds determine what actions are taken when multiple messages trigger the rule within a given time period.

You set thresholds on the Message Delivery panel. You can request that the action associated with the rule be performed before these thresholds are reached, after they are reached, or whenever the rule is triggered, by entering a valid value in the Do Action field.

Note

When a threshold is reached, the value of the Deliver flag is effectively reversed. For example, if the flag is set to NO, messages to which the rule applies are suppressed until the threshold is reached, then delivered to the console. If the flag is set to YES, messages are delivered to the console until the threshold is reached, then suppressed. If the flag is set to LOG, messages are sent to the log until the threshold is reached, then delivered to the console.

By setting the Deliver flag to YES:

You can specify that you do not want to see the same message more than a given number of times (such as 10) within a certain time interval (such as one minute). You do this by entering 10 in the Maximum Number field and 00.01.00 in the Time Interval field.

By setting the Deliver flag to NO:

You can specify that you only want a message displayed if it starts occurring more frequently than usual. You do this by entering a value in the Time Interval field. If more messages of the same kind than the number specified in the Maximum Number field are received within the specified time interval, the messages are displayed—otherwise not.

If you want to see, for example, every fifth occurrence of a message, set the Maximum Number field to four and leave the Time Interval field blank (or set to 0). This indicates that, no matter how long the interval between occurrences of this message, every fifth occurrence of the message is to be displayed. All other occurrences of the message are to be suppressed.

Specifying a Correlation Key

To avoid having to create separate rules for different versions of the same message, you can specify a correlation key on the Message Delivery panel. This enables the rule to keep separate threshold counts for each instance of the correlation key, and avoids the possible suppression of important but uncommon versions of a message.

The correlation key can include:

- A user-defined EventView variable
- A reference to a ZMSG variable, such as &ZMSGWORD3

For example, you might want to limit the number of messages (from a given job) that trigger this rule to ten for every hour. You might, consequently, set the thresholds shown in Figure 17-3.

Figure 17-3. Specifying Message Thresholds

```
SOLVPROD---------------- EventView : Message Delivery ------------------TAPEMON
Command ===>                                                   Function=UPDATE

Deliver ..........+ YES

. Threshold ------------------------------------------------------------------.
|                                                                              |
| Maximum Number .. 10                                                         |
| Time Interval ... 01.00.00                                                   |
| Do Action ......+ ______                                                     |
| Correlation Key   &ZMSGJOBNM                                                 |
|                                                                              |
'------------------------------------------------------------------------------'



 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Modifying Messages

Both message presentation and message text can be modified by specifying the requirements on the Message Modification panel (see Figure 17-4 on page 17-10).

Specifying Replacement Text

By entering replacement text in the Replacement Text field on the Message Modification panel, you can replace the entire message text with an alternative text string. The text can include variables.

Specifying System Message Presentation Parameters

If you want to alter how a message is to be displayed to a system console user, specify the message descriptor code in the Set Descriptor Code field. This code determines the color that the system will use to display the message on a color console, and whether the message is to be non-roll deletable. You can also change the message routing code by specifying a value in the Set Route Code field, which is used by the system to control message delivery.

Specifying SOLVE Message Presentation Parameters

By completing the appropriate fields in the lower box on the Message Modification panel, you can alter how a message is to be displayed to a user. You can also specify whether a console alarm is to be sounded when the message is delivered, and whether the message is to be delivered to monitor class users. The monitor status of a user is set in the user definition and profile.

The example, shown in Figure 17-4, shows that the console alarm is to be sounded when the messages that trigger this rule are delivered.

Figure 17-4. Specifying How to Modify a Message

```
SOLVPROD---------------- EventView : Message Modification --------------TAPEMON
Command ===>                                                   Function=UPDATE

Replacement Text ____________________________________________________________
                 ____________________________________________________________

. Message Presentation -------------------------------------------------------.
|                                                                             |
| Set Descriptor Code ....+ _____________                                     |
| Set Route Code .........+ _____________                                     |
|                                                                             |
'-----------------------------------------------------------------------------'

. SOLVE Message Presentation -------------------------------------------------.
|                                                                             |
| Color ......+ __________  Highlight ...+ _______  Intensity ...+ ______     |
| Monitor? .... ___         Alarm? ....... Y___    NRD? ......... ____        |
| Message Code  __                                                            |
|                                                                             |
'-----------------------------------------------------------------------------'


 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Specifying Actions to be Taken in Response to Messages

On the Message Actions panel, you can specify what response is made to a message; whether a reply is sent, a command issued, or other action taken. Apart from reply text, you can specify:

- System command text, such as: `START STC1.`
- Management Services command text, such as:

  `LOG TEST MSG1 ENCOUNTERED-WORD5=&ZMSGWORD5`
- A process selected from the list of valid processes—enter a question mark in the field to display a list of valid processes.
- An Automation Services command selected from the list of valid commands—enter a question mark in the field to display a list of valid commands. For information about the parameters for Automation Services commands, see Appendix A, *Commands*.

The example, shown in Figure 17-5, loads a new system image in the local region when the rule is triggered.

Figure 17-5. Specifying How to Respond to a Message

```
SOLVPROD---------------- EventView : Message Actions --------------------BACKUP
Command ===>                                                   Function=UPDATE

Reply Text ....... ____________________________________________________________
                   ____________________________________________________________
System Command ... ____________________________________________________________
                   ____________________________________________________________
MS Command ....... ____________________________________________________________
                   ____________________________________________________________

. Automation Actions -------------------------------------------------------.
|                                                  S/B=Browse U=Update L=List |
|      Process     Parameters                                                |
| ___  ________    _________________________________________________________ |
|                  _________________________________________________________ |
|                                                                            |
|      Command     Parameters                                                |
| ___  LOAD        NEWSYS=SOLV NEWVERS=2 MODE=AUTOMATED                      |
|                  _________________________________________________________ |
|                                                                            |
'----------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap                 F11=Panels   F12=Cancel
```

Suppressing Messages

You can reduce message traffic to both the system and the consolidated console by suppressing messages that are not required by operators to perform their tasks. Message suppression does not affect the automated resource monitoring and control functions performed by ResourceView and ServiceView.

Use the following methods to suppress messages:

- Set the Deliver flag from the Message Delivery panel of a message rule. For example, you can specify LOG to suppress messages that trigger the rule from the consoles but still enable them to be logged.
- Use the threshold criteria on the Message Delivery panel of a message rule to suppress redundant messages when multiple messages trigger the rule within a specified time.
- When you have implemented rules for all relevant messages, you can suppress all other messages that are not processed by a rule. To suppress these messages, specify NO or Z in the Default Message Delivery field on the Ruleset Description panel.

  Use the message-learning facility to identify any new messages that have been suppressed. You can then decide whether to create rules for them.

18

Defining Timers

This chapter describes how to use timer rules.

This chapter contains the following topics:

- What Is a Timer Rule?
- Adding and Maintaining Timers
- Displaying Active Timer Rules

What Is a Timer Rule?

A timer rule enables you to schedule an action or actions to perform at a specific time or times of the day, week, month, or year.

A timer rule contains the following information:

- The action or actions to be performed
- A schedule that defines when the action or actions are performed
- Whether catchup is required, if the system running the timer is unavailable when the timer is due to be activated

A timer schedule is similar to the availability map used by resources controlled by the region. You can specify up to 99 schedule items per timer rule, each containing the following information:

- The day of the week, date, and time when the action or actions are to be performed
- Whether the action or actions are to be performed once only, or at regular intervals during a given time period

Adding and Maintaining Timers

If you want to add a timer rule that is very similar to an existing one, you can save yourself having to retype details by copying the existing timer and updating the copy as appropriate. Otherwise, to add a timer rule, you proceed as follows from the EventView Definition Menu:

Step 1. Enter the name of the ruleset with which you want to associate the timer rule and select option **T** - Timer Rules.

The Timer Rule List for the specified ruleset is displayed.

Step 2. Press F4 (Add) to add a timer rule.

The Timer Description panel is displayed. Figure 18-1 on page 18-3 is an example.

Step 3. Complete the fields on this and subsequent timer rule panels as required.

Note

If you enter YES in the Delete on Expiry? field, schedule items that have a full date specified will be purged upon being actioned.

Performing Catchup

When you define a timer, you specify whether catchup is required if a system running the timer is unavailable when the timer is due to be activated.

Note that if you enter YES in the Catchup Required? field on the Timer Schedule panel shown in Figure 18-1, catchup applies to all schedule items entered for this timer.

- If you specify YES, then the scheduled action or actions are performed when the system becomes available, provided that the time specified in the Window field has not elapsed, with the exception of the situation noted below.
- If you specify NO, no belated processing occurs for that timer.

Note

In the case of timers that define actions that are repeated, catchup can be requested. If the specified end time has passed by the time the system running the timer becomes available, the specified action or actions are still performed once. If the system running the timer becomes available part way through the specified time period, the specified action or actions continue at the specified intervals until the specified end time.

Figure 18-1. Adding Timer Schedule Items

```
SOLVPROD------------- EventView : MOUNTS1 Timer Schedule ---------------TAPEMON
Command ===>                                                   Function=ADD

Delete on Expiry? ... NO_

Catchup Required? ... NO_
        Window ...... _____


. Timer Details -------------------------------------------------------------.
|                                                                            |
|     Item  Day   Date          Time       Every   Num   End Time   Status   |
|       1   MON   ___________   16.00.00   00.10   12__  18.00.00   ACTIVE__ |
|       2   TUE   ___________   16.00.00   00.10   12__  18.00.00   ACTIVE__ |
|       3   WED   ___________   16.00.00   00.10   12__  18.00.00   ACTIVE__ |
|       4   THU   ___________   16.00.00   00.10   12__  18.00.00   ACTIVE__ |
|       5   FRI   ___________   16.00.00   00.10   12__  18.00.00   ACTIVE__ |
|       6   SAT   ___________   16.00.00   00.30   4___  18.00.00   ACTIVE__ |
|       7   SUN   ___________   16.00.00   00.30   4___  18.00.00   ACTIVE__ |
|                                                                            |
'----------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=File      F4=Save      F5=NextTmr
 F7=Backward  F8=Forward   F9=Swap     F10=Scrllst  F11=Panels   F12=Cancel
```

Specifying a Catchup Window

If you specify that catchup is required, you can identify the window within which catchup is to be performed. You can specify a value between one minute and 24 hours. If the system running the timer becomes available before the catchup window ends, then catchup is performed, otherwise not.

Specifying Timer Schedule Items

You can enter up to 99 schedule items for a timer. Enter schedule details according to the following definitions:

Day

As well as the abbreviated versions of the days of the week, you can enter shorthand values asterisk (*), W/D, or W/E in this field. If an asterisk is entered, an individual schedule item is created for each of the seven days of the week, with all other values duplicated. If you enter W/D, an individual schedule item is created for each of the five working days of the week. Entering W/E results in the creation of individual schedule items for Saturday and Sunday.

Note

The validation procedure will not accept a value in both the Day and the Date fields; enter a value in one of these fields only.

Date

If you specify a numeric value between 1 and 31 in this field, the timer will be activated on that day of the month each month. For example, if you specify 1, it will be activated on the first day of each month. If, in addition to specifying a day, you also specify the first three characters of a month in the format *dd-mmm*, the timer will be activated on that day of that month each year. If, in addition to specifying a day and a month, you also specify a four-character year value in the format *dd-mmm-yyyy*, the timer will be activated on that day of that month and that year. If you entered YES in the Delete on Expiry? field, schedule items that have a full date specified are purged upon being actioned.

Time

The Time field specifies the time when the action or actions associated with the timer are performed or, if the Every field also contains a value, the time when the action or actions associated with the timer are first performed.

Every

You enter a value in this field if you want the action or actions associated with the timer performed at regular intervals. If you enter a value in this field, you must also enter a value in either the Num or the End Time field. When you complete one of these fields, the other is calculated automatically when validation occurs.

The first time the action or actions associated with the timer will be performed is specified in the Time field—see the preceding definition. To calculate the time when the second occurrence of the action or actions associated with the timer will be performed, the value in the Every field is added to the value in the Time field, and so on.

Num

This field specifies the number of times that the action or actions associated with the timer will be performed. When you enter a value in this field, the value in the End Time field is automatically calculated.

End Time

This field specifies the last permissible time when the regular action or actions associated with the timer are to be performed. When you enter a value in this field, the value in the Num field is automatically calculated.

Status

You can disable an individual timer schedule item by changing the status of that item from active to inactive.

Adding Further Schedule Items

When you have completed the first seven entry lines on the static list displayed initially, you do the following to add further schedule items:

Step 1. Press F10 (Scrllst).

Note

If you are using a 24-line screen, you can type MAX at the Command ===> prompt at this point to maximize screen use and to display just the schedule list, as shown in Figure 18-2.

Step 2. To add a line to the schedule, apply the **R** (Repeat) action to a listed item.

Step 3. Type over the repeated line with the new schedule item details.

Figure 18-2. Full-screen Schedule Display

```
SOLVPROD-------------- EventView : MOUNTS1 Timer Schedule --------------TAPEMON
Command ===>                                                Scroll ===> PAGE

                                                           D=Delete R=Repeat
    Item  Day   Date          Time       Every   Num   End Time   Status
      1   MON                 16.00.00   00.10   12    18.00.00   ACTIVE
      2   TUE                 16.00.00   00.10   12    18.00.00   ACTIVE
      3   WED                 16.00.00   00.10   12    18.00.00   ACTIVE
      4   THU                 16.00.00   00.10   12    18.00.00   ACTIVE
      5   FRI                 16.00.00   00.10   12    18.00.00   ACTIVE
      6   SAT                 16.00.00   00.30   4     18.00.00   ACTIVE
r     7   SUN                 16.00.00   00.30   4     18.00.00   ACTIVE
    **END**
```

Viewing the Next Execution Display for Timer Schedule Items

You can use the F5 (NextTmr) function key on the Timer Schedule panel to display the Next Execution Display panel, shown in Figure 18-3.

This panel displays the next scheduled execution time and date of each timer schedule item in the order that they fall due, as well as all the schedule item details specified in the schedule map.

Figure 18-3. Displaying the Next Execution Times and Dates for a Timer Rule

```
SOLVPROD------------- EventView : Next Execution Display ----------------------
Command ===>                                                   Scroll ===> PAGE

    Item NextDate     NextTime  Day  Date        Time      Every  Num  EndTime
      2  08-AUG-1995  16.00.00  TUE              16.00.00  00.10  12   18.00.00
      3  09-AUG-1995  16.00.00  WED              16.00.00  00.10  12   18.00.00
      4  10-AUG-1995  16.00.00  THU              16.00.00  00.10  12   18.00.00
      5  11-AUG-1995  16.00.00  FRI              16.00.00  00.10  12   18.00.00
      6  12-AUG-1995  16.00.00  SAT              16.00.00  00.30  4    18.00.00
      7  13-AUG-1995  16.00.00  SUN              16.00.00  00.30  4    18.00.00
      1  14-AUG-1995  16.00.00  MON              16.00.00  00.10  12   18.00.00
    **END**









 F1=Help      F2=Split     F3=Exit                   F5=Find      F6=Refresh
 F7=Backward  F8=Forward   F9=Swap
```

Updating Timer Schedule Items

Timer schedule items are updated by typing over the existing values with new values, then pressing F4 (Save). If the item you want to update is not amongst the first seven items displayed on the Timer Schedule panel, use the F10 (Scrllst) function key to scroll down the list until you find it.

Timer schedule items can also be updated from the Active Timers List panel (see the section, *Displaying Active Timer Rules*, on page 18-8).

Deleting Timer Schedule Items

Timer schedule items can be deleted by using the same procedure as for adding schedule entry lines, but applying the **D** (Delete) action to the item to be deleted, instead of the **R** (Repeat) action.

Specifying Actions to be Taken When a Timer Item is Triggered

On the Timer Actions panel, you can specify what response is made when a scheduled timer item is triggered. You can specify:

- System command text, such as: `START STC1`
- Management Services command text, such as:

  `LOG TEST TIMER RULE EXECUTED`
- A process selected from the list of valid processes—enter a question mark in the field to display a list of valid processes
- An Automation Services command selected from the list of valid commands—enter a question mark in the field to display a list of valid commands

Example

The TSO resource is defined to stop automatically at 1900 on weekdays. To warn users of the impending shutdown, you can define a timer that sends a warning message to the users at 1845 on the weekdays (see Figure 18-4).

Figure 18-4. Specifying Timer Actions

```
SOLVPROD---------------- EventView : GRTIMER1 Rule Actions ------------FOGRULE1
Command ===>                                                   Function=COPY

System Command ... SEND 'TSO WILL BE STOPPED IN 15 MINUTES - PLEASE LOG OFF'
                   ____________________________________________________________
MS Command ....... ____________________________________________________________
                   ____________________________________________________________
```

Displaying Active Timer Rules

On the Event Definition Menu, enter **S.T** to display the Active Timers List panel shown in Figure 18-5.

This list displays the date and time of the next scheduled execution of all timer rules that have a status of active *and* are associated with the active ruleset. If you scroll to the right, the schedule item details, as specified on the schedule map, are displayed.

You can browse, update, copy, or delete listed timers.

Figure 18-5. Displaying Active Timers

```
SOLVPROD-------------- EventView : Active Timer Display -----------------------
Command ===>                                                   Scroll ===> PAGE

                                              S/B=Browse U=Update C=Copy D=Delete
    Timer Name  Ruleset  NextDate        NextTime  Short Description
    EVTIMER1    TAPEMON  TUE 27-MAY-1997 17.40.00  TAPEMON ruleset timer 1
    EVTIMER2    TAPEMON  WED 28-MAY-1997 14.00.00  TAPEMON ruleset timer 2
    MSTRULE     MASTER   FRI 30-MAY-1997 12.00.00  MASTER ruleset timer
    **END**
```

19

Message Learning

This chapter describes how to use the message-learning facility.

This chapter contains the following topics:

- What is Message Learning?
- Controlling Message Learning
- Browsing and Updating Learnt Messages
- Generating a Rule for a Learnt Message
- Resetting New Message Indicators
- Deleting All Learnt Messages

What is Message Learning?

The message-learning facility records messages seen by EventView. The facility provides you with a list of all messages encountered during the course of system operation. After you review the initial set of messages, you can reset the new message indicator. Then, when new software is installed, you can easily learn about the new messages.

The facility allows you to:

- List all learnt messages, or all *new* learnt messages
- Display formatted information about listed messages
- Create a message rule from a learnt message
- Use the learnt message list as a prompt list when specifying messages for resource definitions—see the section, *Specifying Messages in a Resource Definition*, on page 5-33

Normally, only the first message that starts with a particular word is learnt. However, since some programs issue diverse messages with the same first word, EventView allows for this possibility. EventView also allows you to learn the minor lines of a multiline message. You can enable these features by entering YES in the Learn Multiple Messages? field on the Message Details panel of a learnt message.

Controlling Message Learning

Message learning can be enabled only if an EventView rule set is loaded with your system image. You control the facility by using the Learn New Messages? field on the Ruleset Description panel of the rule set definition. To access the rule sets, enter the **/EADMIN.R.R** path.

Browsing and Updating Learnt Messages

You can browse and update learnt messages by applying the appropriate action to a listed message. To list learnt messages, you do the following:

Step 1. Select the **L** - Message Learning option from the Event Administration menu.

Step 2. Select either **L** - Learnt Messages (to list all learnt messages) or **N** - New Learnt Messages. See the section, *Resetting New Message Indicators*, on page 19-7 for the definition of a new learnt message.

Step 3. Apply the **B** (Browse) action to a message you want to browse, or the **U** (Update)) action to a message you want to update.

For example, you might want to update the Learn Multiple Messages? field on the Message Details panel, to indicate that you want EventView to learn multiple messages with the same ID.

Step 4. Select the panel you want to browse or update. See the online help for definitions of the fields on the panels.

Figure 19-1 shows an example of the panel sequence when you browse a learnt message.

Figure 19-1. Browsing a Learnt Message

```
SOLVPROD--------- Automation Services : Event Message Learning ----------$RE003
Select Option ===> L

   L   - Learnt Messages
   N   - New Learnt Messages
   R   - Reset 'New Message' Indicators
   D   - Delete All Learnt Messages
   X   - Exit

Message ID ... _____________ ( Optional L N )
```

```
SOLVPROD--------------- EventView : Learnt Message List ---------------$REMSGLN
Command ===>                                                   Scroll ===> PAGE

                                 S/B=Browse U=Update D=Delete GR=Generate Rule
    Message                                                                New
    IEE800D CONFIRM VARY FORCE FOR 4C1 - REPLY NO OR YES                    *
    IEE855I DUMPDS COMMAND RESPONSE                                         *
    IEE889I 16.36.39 CONSOLE DISPLAY 180
    IEE949I 17.36.42 SMF DATA SETS 562
    IEF233A M 380,898811,,PCHUD1TS,COPY1,
    IEF238D DFHSM - REPLY DEVICE NAME OR 'CANCEL'.                          *
s   IEF244I DFHSM DFHSM - UNABLE TO ALLOCATE 1 UNIT(S)                      *
```

```
SOLVPROD--------------- EventView : Panel Display List ------------------------
Command ===> 1                                                 Scroll ===> PAGE

                                      Use 'S' to select Panel/s to be displayed
    Panel Description
    Message Details
    Message Attributes
    Message Presentation
```

```
SOLVPROD---------------- EventView : Message Details ----------------$REMSGLN
Command ===> 3                                                Function=BROWSE

  Message Text ...... IEF244I DFHSM DFHSM - UNABLE TO ALLOCATE 1 UNIT(S)


  Learn Multiple Messages? ...

. Associated Major Line -------------------------------------------------.
|                                                                         |
| Message ID ........ IEF244I                                             |
| Text ............. IEF244I DFHSM DFHSM - UNABLE TO ALLOCATE 1 UNIT(S)  |
|                                                                         |
|                                                                         |
'-------------------------------------------------------------------------'
. Message Presentation --------------------------------------------------.
|                                                                         |
| Descriptor Code ... 6                                                   |
| Route Code ........ 2,3                                                 |
```

```
SOLVPROD--------------- EventView : Message Presentation -------------$REMSGLN
Command ===>                                                  Function=BROWSE

  Message Text ...... IEF244I DFHSM DFHSM - UNABLE TO ALLOCATE 1 UNIT(S)


. SOLVE Message Presentation --------------------------------------------.
|                                                                         |
| Color ....... GREEN      Highlight .... NONE     Intensity .... NORMAL  |
| Monitor? .... NO         Alarm? ....... NO       NRD? ......... NO      |
| Message Code  00                                                        |
```

Generating a Rule for a Learnt Message

If you want to suppress further instances of a message, or to automate the response to the message, you can generate an associated message rule. You do this from the Learnt Message List panel in the following way:

Step 1. Apply the **GR** (Generate Rule) action to a listed item.

The Ruleset List panel is displayed.

Step 2. Select the rule set to which the message rule you want to create is to be added.

The initial panel of the generated message rule is displayed in Add mode. All details stored in the learnt message record that are relevant to message rules have been copied to the message rule record and are displayed in the appropriate fields.

Step 3. Complete the mandatory Short Description field, and complete or update other fields as required.

Step 4. Save the new message rule.

Figure 19-2 shows an example of how to generate a rule, using the same message as is shown in the example in Figure 19-1 on page 19-4.

Figure 19-2. Generating a Message Rule

```
SOLVPROD--------------- EventView : Learnt Message List ---------------$REMSGLN
Command ===>                                                   Scroll ===> PAGE

                                 S/B=Browse U=Update D=Delete GR=Generate Rule
    Message                                                              New
    IEE800D CONFIRM VARY FORCE FOR 4C1 - REPLY NO OR YES                  *
    IEE855I DUMPDS COMMAND RESPONSE                                       *
    IEE889I 16.36.39 CONSOLE DISPLAY 180
    IEE949I 17.36.42 SMF DATA SETS 562
    IEF233A M 380,898811,,PCHUD1TS,COPY1,
    IEF238D DFHSM - REPLY DEVICE NAME OR 'CANCEL'.                        *
gr  IEF244I DFHSM DFHSM - UNABLE TO ALLOCATE 1 UNIT(S)                    *
    IEF290E DFHSM DFHSM RESIN4 NEEDS 1 UNIT(S) FOR                        *
```

```
SOLVPROD----------------- EventView : Ruleset List ---------------------------
Command ===>                                                   Scroll ===> PAGE

                                            Use 'S' to select the required Rule
    Ruleset      Description                                            Status
    ISTMSGS      VTAM Messages                                          ACTIVE
s   MSTRULE      Master ruleset                                         ACTIVE
```

```
- SOLVPROD---------------- EventView : Message Filter -------------------MSTRULE
Command ===> 3                                                 Function=ADD

Ruleset Name ........ MSTRULE                        Rule Status ...+ ACTIVE__
Short Description ... Master ruleset_______________

. Expected Message -------------------------------------------------------------.
|                                        S=ListPanels E=ExtFilter T=TestVars |
|      Message Text  ( WildChar = * )                                   ExtFlt |
| ___  IEF244I DFHSM DFHSM - UNABLE TO ALLOCATE 1 UNIT(S)_____________ NO     |
'-----------------------------------------------------------------------------'
. Execution Conditions ---------------------------------------------------------.
|                                                                               |
| Job Name .... DFHSM___   Rule Priority ........... __   (1 is best)           |
| Job Type ...+ _______    Execute If Not Best Fit?  ___                        |
```

```
SOLVPROD--------------- EventView : Message Modification ---------------MSTRULE
Command ===>                                                   Function=ADD

Replacement Text ____________________________________________________________
                 ____________________________________________________________

. Message Presentation ---------------------------------------------------------.
|                                                                               |
| Set Descriptor Code ....+ 6___________                                        |
| Set Route Code .........+ 2,3_________                                        |
|                                                                               |
'-----------------------------------------------------------------------------'

. SOLVE Message Presentation ---------------------------------------------------.
|                                                                               |
| Color ......+ GREEN____  Highlight ...+ NONE___  Intensity ...+ NORM          |
| Monitor? .... NO_        Alarm? ....... NO_      NRD? ......... NO_           |
| Message Code  00                                                              |
|                                                                               |
'-----------------------------------------------------------------------------'
```

Resetting New Message Indicators

If you want to differentiate between messages learnt before and after a certain date, you can reset the new message indicators for the entire Message Learning database on that date. You might also want to reset the new message indicators after you review and create rules for the current learnt messages.

At a later date, you can select the **N** - New Learnt Messages option from the Event Message Learning menu to list only those messages that have been learnt since you reset the new message indicators (for example, since the last review).

If you list all learnt messages, the messages that are flagged as new messages are identified by the presence of an asterisk (*) in the New column on the Learnt Message List panel (shown in the upper panel in Figure 19-2 on page 19-6).

Deleting All Learnt Messages

To avoid accumulating too many messages, you might want to purge all messages after you have viewed those messages that interest you and generated appropriate rules. By selecting option **D** - Delete All Learnt Messages from the Event Message Learning menu shown in Figure 19-1 on page 19-4, you can purge *all* learnt messages from the Message Learning database.

Caution

Purged messages cannot be recovered.

The AUTOTABLES parameter group controls the size of the table that stores the learnt messages.

20

Defining and Maintaining Message Profiles for the Consolidated Console

The console consolidation facility consolidates console message traffic from multiple systems onto a single screen (known as a *consolidated console*). Operators can thus view those messages from a single console. You create *message profiles* that contain criteria to identify and classify messages. If a user requests messages for a given message profile, all messages that match the criteria of that profile are displayed on that user's consolidated console.

This chapter contains the following topics:

- How Console Consolidation Works in a Multisystem Environment
- Implementing Message Profiles
- Accessing the Message Profile Definitions
- Defining a Message Profile
- Changing the Activation Status of a Message Profile
- Activating Message Profiles
- Maintaining Message Profile Definitions

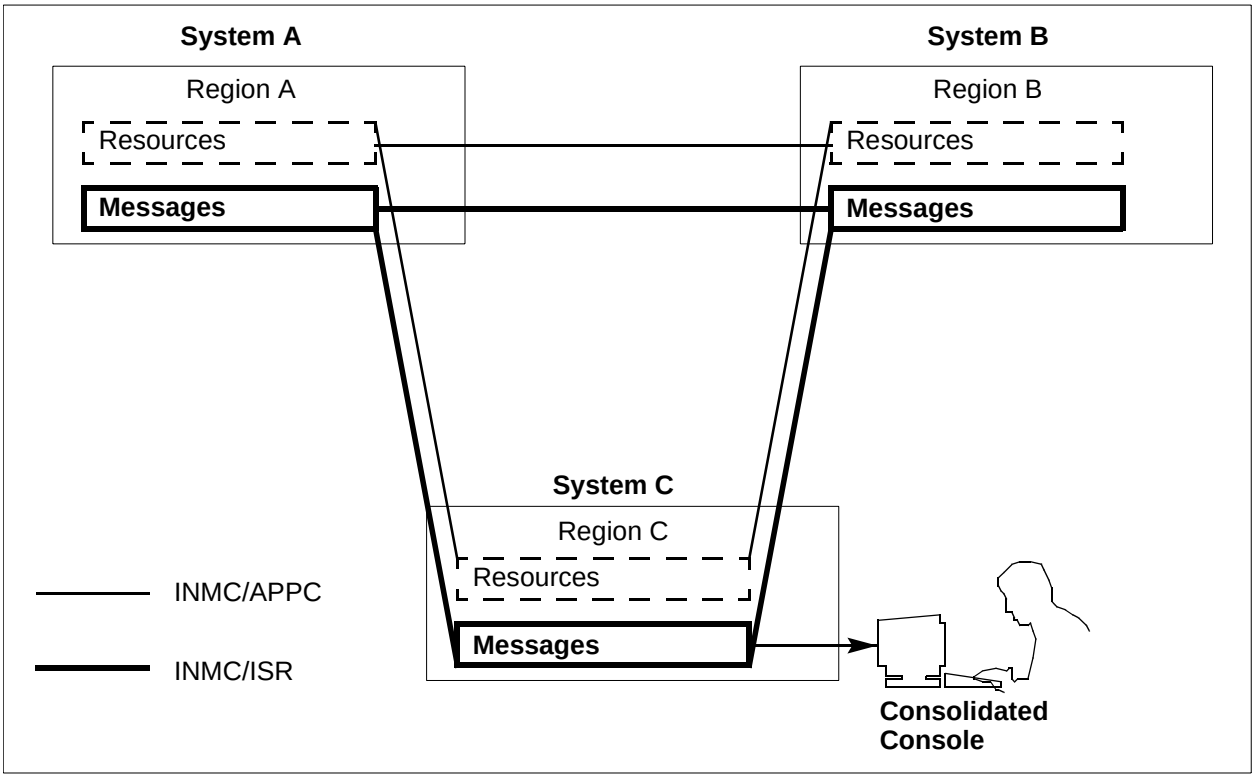
Note

Multisystem message visibility is available only at consolidated consoles in focal point regions. In subordinate regions, only local messages are visible.

How Console Consolidation Works in a Multisystem Environment

Multisystem support at the message level provides for the distribution of messages to consolidated consoles in focal point regions in the multisystem environment. Each region communicates with other connected regions by using Inter-Management Services Connection (INMC)/Inter-System Routing (ISR) links (Figure 20-1).

Figure 20-1. Multisystem Links for Consolidated Consoles



Each region has an ISR link manager. The ISR link manager is started up as part of region initialization. The ISR link is active but disabled until a user starts console consolidation. The ISR link manager enables message flow across the link based on requests from users for messages that match specific message profiles.

The user profile determines the messages seen on a consolidated console. The ISR link manager suppresses those messages that are not required, thus reducing the amount of message flow. A user who has not been assigned message profiles or has all the assigned message profiles disabled sees no messages on the consolidated console.

Implementing Message Profiles

You can define profiles that capture different types of messages. When you create or make changes to a message profile, the data is automatically distributed to the knowledge bases in connected regions.

Note

You can define message profiles in focal point regions only. However, the defined profiles are available to subordinate regions through knowledge base synchronization.

A message profile contains the following criteria types that determine which messages are received by a consolidated console:

- The system from which the message comes
- The ID (or the first word) of the message
- The job for which the message is generated
- The message routing and descriptor codes
- The message types and levels, and the types and classes of job for which the message is generated

A message profile must use at least one criterion from the last four criteria types.

Each profile has a status that determines whether it can be activated for use. Profiles must be activated, either by you or automatically during region startup, before they can be used. After you define the profiles, you need to activate them for use by the operators.

Rules for Defining and Using Message Profiles

This section contains rules about entering data on panels and about how to get the best results when defining message profiles.

Create New Message Profiles in a Single System First

Create a new message profile to select messages from one system only, using selection criteria that are unique to that system. For example, if each system uses different message classes, specify a message class that is unique to your system. When this profile is working successfully in one system, you can copy it into a new profile for other systems whose messages you want to be monitored.

Use Unique Message Profile Names and IDs

Message profiles are identified by unique profile names and IDs. When messages are captured, they are associated with a specific profile ID. The profiles replace the message routing codes corresponding to the IDs as the means for the region to direct relevant messages to operators. An operator who wants to receive specific messages on a consolidated console enables the relevant profiles. Alternatively, if the operator always wants to see consolidated messages for certain profiles, the operator can specify this information in the user profile. See Chapter 21, *Monitoring Messages*, for information about how to use message profiles.

Caution

A profile acts on messages after they are processed by EventView message rules. For example, if a rule changes the routing code and you want to capture the message, ensure that the profile ID corresponds to the changed routing code.

You cannot include special characters (for example, _, -, (,), and ~) or spaces in a profile name.

Using Wildcards

Use wildcard characters to represent character patterns at particular positions in a character string. The supported wildcard characters are as follows:

- *, representing any character as follows:
  - If the * is at the beginning of or embedded in a character string, it represents one character.
  - If the * is at the end of a character string, it represents any number of characters.

  You *cannot* use an * by itself. In the following example, messages are selected for any system that starts with the letters EAST:

  ```
  Systems to be Included
  EAST*
  ```

- #, representing one numeric character. In the following example, messages are selected for systems EAST0 through EAST9:

  ```
  Systems to be Included
  EAST#
  ```

- @, representing one alpha character. In the following example, messages are selected for systems EAST0A through EAST9Z:

  ```
  Systems to be Included
  EAST#@
  ```

Type as many characters as necessary to select the required information.

Resetting a Wildcard Character to Its Literal Sense

If you want to use a wildcard character in the literal sense, precede the character by a backslash (\). For example, ABC### matches any value that starts with ABC followed by three numeric characters; ABC##\# enables you to match a value that starts with ABC followed by two numeric characters and ending in a # character.

Using Ranges

Use a colon (:) to specify ranges. In the following example, messages are selected for systems EAST0, EAST1, and EAST2:

```
Systems to be Included
EAST0:EAST2
```

The character string on each side of the colon must conform to the following rules:

- The strings must be of equal length.

  Note

  The backslash (\) is regarded as one character when the length of the string is calculated.

- The asterisk (*) wildcard character can only be used at the end of a string.

Including and Excluding Information

In each profile, you must specify the criteria that determine the messages to display on the consolidated console. Most panels have *inclusion fields* and *exclusion fields*, or allow you to specify N(o) or Y(es), according to whether you want to include or exclude messages with certain attributes. The rules for including and excluding messages are as follows:

- If you leave all the fields on a panel blank, the criteria specified on the other panels determine what messages are displayed on the consolidated console. For example, if you leave the System Specification panel blank, messages from all connected regions are potentially available for display.

  However, if you do *not* specify any criteria in a message profile (that is, if you leave the criteria fields on all the panels blank), the profile receives *no* messages.

- If you specify both inclusion and exclusion values for a particular criterion, the inclusion values take priority. The exclusion values are then applied to the resulting set of included messages.

  For example, using message ID as a criterion, if you want to include all message IDs except the IDs starting with AAA111, you can use the following inclusion and exclusion values:

Inclusion values	A*:9*
Exclusion values	AAA111*

- Messages are selected for display only if they meet the criteria specified on all panels. For example, if you specify YES in the Sess field on the Message Job Specification panel, only SESS type messages are displayed, even if messages of other types meet the criteria specified on the other panels.

- Items selected with N or Y have an OR relationship. For example, if you include routing codes 1, 2, and 11, messages that have a routing code of 1 *or* 2 *or* 11, *or* a combination of these codes, are displayed if they satisfy the other criteria specified in the profile.

- If you complete an exclusion field or specify N, a message that meets this criterion is not displayed, even if it satisfies all the inclusion criteria specified in the profile.

Note

The consolidated console does not receive messages suppressed by EventView rules.

Accessing the Message Profile Definitions

Enter the **/EADMIN.C.M** path to access the list of message profile definitions.

The Message Profiles panel is displayed. This panel lists all the message profiles in the knowledge base. You can enter action codes to perform actions on existing message profiles, or press F4 (Add) to add a new profile.

Defining a Message Profile

To add a message profile, press F4 (Add) from the Message Profiles panel. You define the profile by using the following panels:

- Profile Details—You must complete this panel. The panel enables you to identify the profile.
- System Specification—This panel enables you to use the system associated with a message as a selection criterion.
- Message Specification—This panel enables you to use the message ID as a selection criterion.
- Job Name Specification—This panel enables you to use the job associated with a message as a selection criterion.
- OS Codes Specification—This panel enables you to use the routing and descriptor codes associated with a message as selection criteria.
- Message Job Specification—This panel enables you to use the message type and level, and the job type and class associated with a message as selection criteria.

You can create a profile to capture particular messages (for example, all tape mount messages) or to capture messages for particular jobs (for example, all production CICS jobs). You do not need to complete every panel for most profile definitions. However, you must complete one of the criteria panels. If you leave all the criteria panels blank, the profile blocks all messages.

Defining the Profile Details

Use the Profile Details panel to identify the message profile. You must specify the profile name, ID, and description. All profile panels contain this information.

You *cannot* use the value 2 as the profile ID. This value is reserved for internal use by the Automation Services software.

Only profiles that have IDs corresponding to those set for a parameter in the CCONSOLIDATN parameter group are available for use in the local region. The parameter might exclude certain IDs. To display the value of the parameter, enter the **/ICS** shortcut to access the list of parameter groups and browse the CCONSOLIDATN parameter group.

The Profile Details panel also contains information about the profile status, whether to profile for solicited messages, and a system-supplied history of when the profile was created and last updated. Only profiles with an ACTIVE status can be activated for use.

Specifying the System Criteria

From the Profile Details panel, press F8 (Forward) to display the System Specification panel. You can specify the systems for which messages are to be captured.

The values you use in the SYSTEMs fields are either the system management facilities (SMF) ID or the region domain ID. The value type is indicated at the bottom of the panel as SMFID or NMDID respectively, and is set in the CCONSOLIDATN parameter group.

Note

The SMFID type indicates different values for VM and VOS3 systems. For the differences, see Help.

The criteria can be specific, generic, or in a range.

Leave the fields blank to allow messages for all the connected systems to be captured. These messages are captured only if other criteria are specified in the profile and the messages satisfy those criteria.

Specifying the Message ID Criteria

From the System Specification panel, press F8 (Forward) to display the Message Specification panel. You can specify the IDs (or generic IDs, for example, $HASP*) of the messages you want to capture. The message ID is the first word of a message.

The values can be specific, generic, or in a range.

Leave the fields blank to allow messages with any ID to be captured. These messages are captured only if other criteria are specified in the profile and the messages satisfy those criteria.

Specifying the Job Criteria

From the Message Specification panel, press F8 (Forward) to display the Job Name Specification panel. You can name the jobs for which messages are to be captured.

The values can be specific, generic, or in a range.

Leave the fields blank to allow messages for all jobs to be captured. These messages are captured only if other criteria are specified in the profile and the messages satisfy those criteria.

Specifying the System Codes Criteria

From the Job Name Specification panel, press F8 (Forward) to display the OS Codes Specification panel. You can specify the routing and descriptor codes assigned to messages that are to be captured. Messages may contain one or a combination of the codes you specify. Messages are not selected if they contain codes that you exclude specifically.

You can exclude certain codes by typing **N** under the codes, include certain codes by typing **Y** under the codes, and leave the other code fields blank. A message containing any of the included codes is selected unless the message also contains an excluded code.

Leave the fields blank to allow messages that contain any routing and descriptor codes to be captured. These messages are displayed on the consolidated console if other criteria are specified in the profile and the messages satisfy those criteria.

Specifying the Message Type, Level, and Job Criteria

From the OS Codes Specification panel, press F8 (Forward) to display the Message Job Specification panel. You can specify the message types, and message levels, job types, and job classes assigned to messages that are to be captured.

You can either include or exclude certain items within each criteria type, but not both (except for the Broadcast field under Message Levels). For example, if you want to accept immediate action messages but not broadcast messages, specify Y in the Immediate Action field and N in the Broadcast field.

Leave the fields blank to allow messages of any type, level, job type, or job class. These messages are displayed on the consolidated console if other criteria are specified in the profile and the messages satisfy those criteria.

Message Types

Message types correspond to the operands of the MONITOR or STOPMN system commands. For example, messages generated because of the MONITOR SESS command have the SESS type.

Message Levels

Message levels indicate the relative importance of a message.

Note

The broadcast level has precedence over all other message criteria. If broadcast messages are allowed, the message profile passes all broadcast messages irrespective of the other criteria.

Job Types

Job types are as follows:

Type	Description
Job	Indicates a batch job.
STC	Indicates a started task job.
TSU	Indicates a TSO user job.
Unknown	Indicates a job that is not one of the above.

Job Classes

The job class is assigned by the CLASS parameter of the JOB JCL statement.

Example 1—Profiling Specific Messages

In this example, the organization has two branches: an eastern branch and a western branch. You want to create a profile to capture all tape mount messages for all the production systems running in the eastern data center, but do not want to capture messages for development jobs. The job classes assigned to tape mount requests are 1, 2, and 3.

From the Message Profiles panel, you press F4 (Add) to add a profile. A Profile Details panel is displayed. You can press F8 (Forward) to scroll forward through the panels or enter the panel index number at the Command ===> prompt to select a specific panel. (Pressing F11 (Panels) displays of a list of panels and panel index numbers.)

Use the following panels in this example:

- Profile Details
- System Specification
- Message Specification
- Job Name Specification
- Message Job Specification

On the Profile Details panel, type a unique profile name (TAPEMOUNTS), a unique ID (127), a description of the profile (Tape Mounts for Eastern Production Jobs), and assign a status. Assign a status of ACTIVE so that the profile can be activated.

Figure 20-2 shows the completed Profile Details panel.

Figure 20-2. Example—Profile Details Panel

```
SOLVPROD----------------- EventView : Profile Details ------------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... TAPEMOUNTS       ID ..... 127 (1 - 128)                             |
| Description .... Tape Mounts for Eastern Production Jobs___________________ |
+-----------------------------------------------------------------------------+
+ Profile Status ----------------------------------------------------------+
| Profile Status ... ACTIVE__ (Active/Inactive)                               |
+-----------------------------------------------------------------------------+
+ Include Solicited Messages? ---------------------------------------------+
| Solicited Type ... NO___    (No, Other, Nothr, Yes, All)                    |
+-----------------------------------------------------------------------------+

+ History ------------------------------------------------------------------+
|                                                                             |
| Profile Created           Profile Last Updated      Profile Status Updated  |
| Userid  USER01            Userid                    Userid                  |
| Date .. THU 22-MAY-1997   Date ..                   Date ..                 |
| Time .. 14.48.27          Time ..                   Time ..                 |
+-----------------------------------------------------------------------------+

 F1=Help      F2=Split     F3=File      F4=Save
              F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Press F8 (Forward) to scroll forward to the System Specification panel. You do not want to capture messages for any western branch systems, so you complete the exclusion fields. All western branch systems start with the letters WST, so WST* is typed to exclude all western branch systems.

Figure 20-3 shows the completed System Specification panel.

Figure 20-3. Example—System Specification Panel

```
SOLVPROD--------------- EventView : System Specification ---------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... TAPEMOUNTS       ID ..... 127 (1 - 128)                             |
| Description .... Tape Mounts for Eastern Production Jobs                     |
+------------------------------------------------------------------------------+
+------------------------------------+ +------------------------------------+
| Systems to be Included             | | Systems to be Excluded             |
+------------------------------------+ +------------------------------------+
| ________________ ________________ | | WST*____________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
+------------------------------------+ +------------------------------------+

Note : Identify the systems by using SMFID


 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Press F8 (Forward) to scroll forward to the Message Specification panel. You only want to display IEF233A messages, which are requests for tape mounts, so you complete the inclusion fields.

Figure 20-4 shows the completed Message Specification panel.

Figure 20-4. Example—Message Specification Panel

```
SOLVPROD-------------- EventView : Message Specification ---------MCPROFIL-0000
Command ===>                                                   Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... TAPEMOUNTS       ID ..... 127 (1 - 128)                             |
| Description .... Tape Mounts for Eastern Production Jobs                     |
+------------------------------------------------------------------------------+
+------------------------------------+ +------------------------------------+
| Message IDs to be Included         | | Message IDs to be Excluded         |
+------------------------------------+ +------------------------------------+
| IEF233A__________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
+------------------------------------+ +------------------------------------+



 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap     F10=Scrllst  F11=Panels   F12=Cancel
```

Press F8 (Forward) to scroll forward to the Job Name Specification panel. You do not want to capture messages for development jobs. All development jobs in the eastern branch start with the letters DEV, so DEV* is typed in an exclusion field.

Figure 20-5 shows the completed Job Name Specification panel.

Figure 20-5. Example—Job Name Specification Panel

```
SOLVPROD-------------- EventView : Job Name Specification --------MCPROFIL-0000
Command ===>                                                   Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... TAPEMOUNTS       ID ..... 127 (1 - 128)                             |
| Description .... Tape Mounts for Eastern Production Jobs                     |
+------------------------------------------------------------------------------+
+------------------------------------+ +------------------------------------+
| Job Names to be Included           | | Job Names to be Excluded           |
+------------------------------------+ +------------------------------------+
| _______________ _______________ | | DEV*___________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
| _______________ _______________ | | _______________ _______________ |
+------------------------------------+ +------------------------------------+



 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap     F10=Scrllst  F11=Panels   F12=Cancel
```

Enter **6** at the Command ===> prompt to display the Message Job Specification panel. Here you want to capture messages for jobs only, in job classes 1 (for jobs that need one tape mounted), 2 (for jobs that need two tapes mounted), and 3 (for jobs that need three tapes mounted).

Figure 20-6 shows the completed Message Job Specification panel.

Figure 20-6. Example—Message Job Specification Panel

```
SOLVPROD------------ EventView : Message Job Specification -------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... TAPEMOUNTS       ID ..... 127 (1 - 128)                             |
| Description .... Tape Mounts for Eastern Production Jobs                     |
+------------------------------------------------------------------------------+
+------------------------------------------------------------------------------+
| Message Types        ( Y =Include, N =Exclude, Blank =Don't care )           |
|  Jobnames .. ___   Status .. ___   Active .. ___   Sess .. ___               |
|                                                                              |
| Message Levels       ( Y =Include, N =Exclude, Blank =Don't care )           |
|      WTOR .. ___  Immediate Action .. ___  Critical Eventual .. ___          |
|  Eventual .. ___     Informational .. ___          Broadcast .. ___          |
|                                                                              |
| Job Types            ( Y =Include, N =Exclude, Blank =Don't care )           |
|       Job .. YES   STC .. ___   TSU .. ___   Unknown .. ___                  |
|                                                                              |
| Job Classes          ( Y =Include, N =Exclude, Blank =Don't care )           |
|              ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789                            |
| (A-Z,0-9) ..                             YYY                                 |
+------------------------------------------------------------------------------+
 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward               F9=Swap                  F11=Panels   F12=Cancel
```

Example 2—Profiling Messages for Specific Jobs

In this example, you want to create a profile to capture messages for certain CICS jobs on the production systems in both the eastern and the western branches. The branches use only one test system, ETST. You assign a status of INACTIVE, as you do not want the profile to be used immediately. You only want to capture messages that have routing codes of 1, 2, or 11.

From the Message Profiles panel, you press F4 (Add) to add a profile. A Profile Details panel is displayed. You can press F8 (Forward) to scroll forward through the panels or enter the panel index number at the Command ===> prompt to select a specific panel. (Pressing F11 (Panels) displays a list of panels and panel index numbers.)

Use the following panels in this example:

- Profile Details
- System Specification
- Job Name Specification
- OS Codes Specification

Figure 20-7 shows the completed panels.

Figure 20-7. Example—Profiling for Messages of Specific Jobs

```
SOLVPROD----------------- EventView : Profile Details ------------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... CICSMESSAGES     ID ..... 126 (1 - 128)                             |
| Description .... Messages for all production CICS Jobs_____________________ |
+-----------------------------------------------------------------------------+

+ Profile Status ----------------------------------------------------------+
| Profile Status ... INACTIVE (Active/Inactive)                             |
```

```
SOLVPROD--------------- EventView : System Specification ---------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... CICSMESSAGES     ID ..... 126 (1 - 128)                             |
| Description .... Messages for all production CICS Jobs                       |
+-----------------------------------------------------------------------------+
+-------------------------------------+ +-------------------------------------+
| Systems to be Included              | | Systems to be Excluded              |
+-------------------------------------+ +-------------------------------------+
| _________________ _________________ | | ETST_____________ _________________ |
```

```
SOLVPROD-------------- EventView : Job Name Specification --------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... CICSMESSAGES     ID ..... 126 (1 - 128)                             |
| Description .... Messages for all production CICS Jobs                       |
+-----------------------------------------------------------------------------+
+-------------------------------------+ +-------------------------------------+
| Job Names to be Included            | | Job Names to be Excluded            |
+-------------------------------------+ +-------------------------------------+
| CICSPRD1:CICSPRD9 CICSTST*__________ | | CICSPRD4__________ _________________ |
```

```
SOLVPROD------------- EventView : OS Codes Specification ---------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... TAPEMOUNTS       ID ..... 127 (1 - 128)                             |
| Description .... Tape Mounts for Eastern Production Jobs                     |
+-----------------------------------------------------------------------------+
+-----------------------------------------------------------------------------+
| Routing Codes    ( Y =Include, N =Exclude, Blank =Don't care )              |
|                    1         2         3         4         5         6      |
|           123456789012345678901234567890123456789012345678901234567890123456 |
|   1-64 => YY        Y                                                       |
```

Now that the panels are complete, press F3 (File) to save the definition.

Example 3—Profiling All Messages

Note

This example is for illustration only. In a multisystem environment, if you have not implemented EventView message rules to provide a high level of message suppression, using this message profile can result in a very high volume of message flow to the consolidated console.

In this example, you want to create a profile to capture the messages for all connected systems. You allow all messages by excluding a system that is not part of the network. Figure 20-8 shows an example, where DMMY is the excluded system.

Figure 20-8. Example—Profiling for All Messages

```
SOLVPROD--------------- EventView : System Specification ---------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... ALLMESSAGES      ID ..... 127 (1 - 128)                             |
| Description .... All messages                                                |
+------------------------------------------------------------------------------+
+------------------------------------+ +------------------------------------+
| Systems to be Included             | | Systems to be Excluded             |
+------------------------------------+ +------------------------------------+
| ________________ ________________ | | DMMY____________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
| ________________ ________________ | | ________________ ________________ |
+------------------------------------+ +------------------------------------+

Note : Identify the systems by using SMFID


 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Example 4—Profiling Messages for a Particular System

In this example, you want to create a profile to capture the messages for a particular system. Figure 20-9 shows an example, where ETST is the system whose messages you want to monitor.

Figure 20-9. Example—Profiling for Messages on a Particular System

```
SOLVPROD--------------- EventView : System Specification ---------MCPROFIL-0000
Command ===>                                                  Function=UPDATE

+---------------------- Message Classification Profile -----------------------+
| Name ... EASTTESTMSGS     ID ..... 127 (1 - 128)                             |
| Description .... All messages for the EASTTEST system                        |
+------------------------------------------------------------------------------+
+-------------------------------------+ +-------------------------------------+
| Systems to be Included              | | Systems to be Excluded              |
+-------------------------------------+ +-------------------------------------+
| ETST_____________ _________________ | | _________________ _________________ |
| _________________ _________________ | | _________________ _________________ |
| _________________ _________________ | | _________________ _________________ |
| _________________ _________________ | | _________________ _________________ |
| _________________ _________________ | | _________________ _________________ |
| _________________ _________________ | | _________________ _________________ |
| _________________ _________________ | | _________________ _________________ |
+-------------------------------------+ +-------------------------------------+

Note : Identify the systems by using SMFID


 F1=Help      F2=Split     F3=File      F4=Save
 F7=Backward  F8=Forward   F9=Swap                  F11=Panels   F12=Cancel
```

Changing the Activation Status of a Message Profile

A message profile must have an ACTIVE status before it can be activated for use. If you only want to change the Profile Status Field for a profile, change the value directly from the Message Profiles panel.

To change the value to ACTIVE, type **A** beside all the profiles you want to update and press ENTER. The value in the Status column for the profiles changes to ACTIVE.

To change the value to INACTIVE, type **I** beside all the profiles you want to update and press ENTER. The value in the Status column for the profiles changes to INACTIVE.

Activating Message Profiles

Note

The message profile activation process can halt the region for a short period of time. After this period, the region continues from where it left off, without loss of control or data. However, delays might occur in responses to system activities. Unless the activation of the message profiles is of a high priority, perform this task when the system is not busy.

After you have created or updated message profiles, you must activate (load) them in one of the following ways in each of the linked regions before they can be used:

- Select the **A** option on the System Console Consolidation panel, or enter the **/EADMIN.C.A** path (available to focal point regions only).
- Enter **ACTIVATE** at the Command ===> prompt on the Message Profiles panel (available to focal point regions only). To display the panel, enter the **/EADMIN.C.M** path.
- Action the CCONSOLIDATN parameter group (available to both focal point and subordinate regions). To display the list of parameter groups, enter the **/ICS** shortcut.

A region only activates profiles with a status of ACTIVE.

Profiles with a status of ACTIVE also become active automatically whenever the region is started. Profiles with a status of INACTIVE are not activated when the region is started.

Message Profile Size Considerations

Activation of message profiles can fail if the total size of the profiles to be loaded is too large. If the problem occurs, a message is generated to indicate by how much the size should be reduced. The ID of the message is either RMCCST11 or RMINWI36. Refer to the message online help for information about how to correct the problem.

Maintaining Message Profile Definitions

In a focal point region, you can browse, update, copy, and delete message profile definitions from the Message Profiles panel.

The *Automation Services Administrator Guide* describes how to assign message profiles to individual users.

21

Monitoring Messages

The console message consolidation facility enables authorized users to view console message traffic from multiple systems on a single console (referred to as a *consolidated console*). An authorized user can create *message profiles* that contain criteria to identify and classify messages. When you use a consolidated console, you use message profiles to select the messages for viewing. All messages that match the criteria of the profiles are displayed on your consolidated console.

Note

The facility is fully functional in focal point regions only. In subordinate regions, only local console message traffic is visible.

This chapter contains the following topics:

- Message Monitor.
- Consolidated Console Setup Requirements.
- Accessing the Consolidated Console.
- Using Message Profiles to Select the Messages to Be Monitored.
- Replying to a WTOR Message From the Consolidated Console
- Leaving the Consolidated Console.

Message Monitor

The message monitor lets you monitor messages in and enter commands to operate a network. The console is accessed from the Monitoring and Control Menu by selecting the M option and can be profiled as a consolidated console.

The monitor uses a formatted display panel called a *window* to provide an environment for executing commands. Your command results are returned to the display panel, along with other system information, to provide a *console* function.

The message monitor is based on the Operator Console Services (OCS). A brief description of the monitor is given here. The *Management Services User's Guide* provides description of OCS.

Operator Profile

Because of the system controls it provides, access to the message monitor is usually restricted to authorized users. Therefore, your activities from the monitor are limited to some extent by the level of authority prescribed for you in your user ID definition.

Each authorized user has a set of attributes, privileges, and options collectively termed the *operator profile*.

Certain attributes of this profile, such as the command authority level, are specified as part of your user ID definition and should not be changed. Other attributes, such as those that customize the monitor display, can be changed by you. To access your profile, enter the **=U.UP** path.

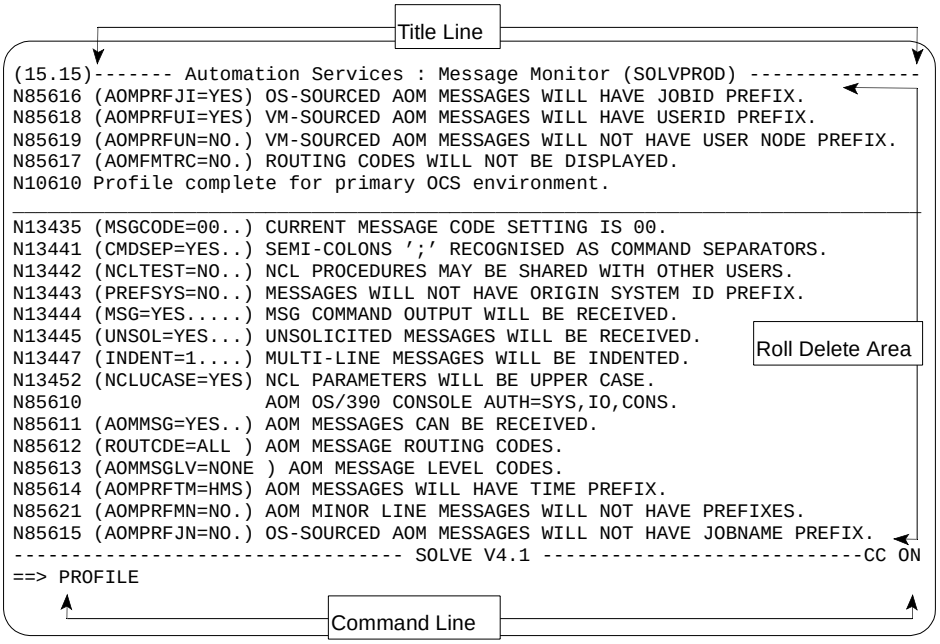
Note

Customized values (that is, values other than the default) are set for the message monitor only if the $RMCCOCS procedure is run. The procedure is specified in the user ID definition.

Window

The message monitor window has two distinct activity areas: a one-line *command input area* at the bottom of the window, and an output message display area called the *roll delete area*, which occupies the remaining space above the command line. The size of the roll delete area will therefore depend on how large you make the window. Figure 21-1 shows the layout of a window.

Figure 21-1. Message Monitor Window Layout



Command Line

The command line is the bottom line of the message monitor. Its width depends on the width of the window, and can be much less than the width of the screen if a vertical split is used.

The command line is the only field where input is allowed. The cursor is automatically positioned at the left hand end of the line when the window is first displayed. To *enter* a command, the cursor must be located in the command line when you press ENTER. In the example shown in Figure 21-1, the PROFILE command has been entered on the command line.

Roll Delete Area

The resulting messages from commands entered on the command line are reported in the roll delete area with other unsolicited information.

Messages are written from top to bottom on the screen. The next line for use is filled by a roll-delimiter line. When the display area is full, new output messages wrap back to the top of the screen, overwriting the oldest displayed messages first. The roll-delimiter line separates the oldest displayed messages from the newest displayed messages. Messages below the line are the oldest; messages above the line are the newest.

Non-roll Delete Area

Most messages displayed on a message monitor are classified as roll delete messages. This means they display once and eventually roll off the screen as subsequent messages arrive and overwrite them.

Non-roll delete (NRD) messages are messages which display until deleted by a *Delete Operator Message* (DOM) function. NRD messages within an message monitor window change the standard display format by introducing a *non-roll delete area* at the top of the normal *roll delete area*.

Non-roll delete messages accumulate in the non-roll delete area. The maximum size of this area is prescribed by your operator profile.

The non-roll delete area is separated from the roll delete area by a delimiter line.

Title Line

The top of the non-roll delete display area is reserved for a title. In the example shown in Figure 21-1 on page 21-3, the title is Automation Services : Message Monitor (SOLVPROD).

Controlling the Roll Delete Message Display Flow

Sometimes, a large number of messages are sent to the screen within a very short time, causing the display to roll the messages faster than you can read them. Use the following methods to suspend the display of new messages temporarily:

- AUTOHOLD—your message monitor window automatically freezes when a message fills the last line of the roll delete area and there are messages queued to wrap back to the top of the display area. The frozen window displays the caption AUTOHOLD above the command line. No further messages appear until you press ENTER. You can use the AUTOHOLD command to change this display characteristic. The *Command Reference* describes the AUTOHOLD command.
- HOLDING—to stop the flow of messages to the roll delete area at any time, press ENTER to freeze the window (make sure there is nothing in the command line—a *null entry*). The frozen window displays the caption HOLDING immediately above the command line. No further messages appear until you press ENTER.

Note

If more unsolicited messages arrive while the window is in the AUTOHOLD or HOLDING mode, the caption above the command line changes to MSG QUED, and the terminal alarm sounds.

Handling Non-Roll Delete (NRD) Messages

There are two categories of NRD messages:

- NRD=YES—these messages are held in the NRD queue, and are retained until explicitly deleted from the queue.
- NRD=OPER—these messages are not held in the NRD queue, and are displayed only at individual message monitor windows until deleted from the window.

Deleting NRD Messages

An NRD message is deleted automatically when one of the following conditions is satisfied:

- The condition to which the NRD message refers is satisfied
- An NCL process deletes the NRD message by issuing an &NRDDEL NCL statement
- The NCL process with which the NRD message is associated stops

To delete NRD messages manually, use the PURGE command. The *Command Reference* manual describes the PURGE command.

Removing NRD Messages (NRD Cursor Deletion)

To remove an NRD message from the message monitor, move your cursor to the line containing the NRD message and press ENTER. The NRD message disappears. This is *NRD Cursor Deletion*. Removing NRD messages in this way provides room for pending NRD messages.

Note

The NRD messages with the NRD=YES attribute are not deleted from the NRD queue, only from your monitor. The NRD messages with the NRD=OPER attribute are deleted and cannot be restored.

Restoring NRD Messages

The NRD messages in the NRD queue can be restored by entering the NRDRET command. The oldest NRD messages are returned to the non-roll delete area first, until the area has expanded to its maximum size.

Prefixing Messages With the System Name

Use the following command to specify whether you want your messages prefixed with the originating system name:

```
PROFILE        AOMPRFSN={NO | YES}
```

For example, to prefix the displayed messages with the system name, enter PROFILE AOMPRFSN=YES.

The changed value is valid for the current session only. If you want to specify a value to use whenever you access the message monitor as a consolidated console, specify the value at the Message Monitor Message Formatting panel of your user profile.

Consolidated Console Setup Requirements

To use the consolidated console, you must be authorized to use the message monitor and AOM, and authorized to receive AOM messages on the monitor. This information is specified in your user ID definition.

In addition, the message monitor must be profiled correctly to receive the relevant messages.

Authorization Requirements

Your authority for using the consolidated console should have been set up by the administrator.

If the User Access Maintenance Subsystem (UAMS) is used to manage authorization, enter the **/UAMS.B** path to browse your user ID definition.

The authorization requirements are as follows:

UAMS Panel (Page Number)	Field	Value
User Access (3)	Operations Management	Y
	Operator Console Services	Y
OCS Details (6)	Initial OCS Command	-$RMCCOCS
AOM General Details (11)	AOM Message Receipt	Y
	Console Routing Codes	ALL
	Message Level Screening	ALL

Monitor Profile Requirements

To enable you to receive messages on a consolidated console, ensure that the following fields on the Message Monitor Message Receipt panel of your user profile have the values Y:

- Receive Unsolicited Messages
- Receive System Messages

To access your user profile, enter the **=U.UP** path.

Accessing the Consolidated Console

From the primary menu, enter **M.M** to access the message monitor. If the lower right of your screen is not displaying CC ON or CC PND, enter **CCON** to change the monitor to a consolidated console. If an RMCCOC07 message is displayed or if the status is CC PND, your console is unable to receive system messages because your profile is not suitable for the consolidated console. See the next section about how to correct your profile.

The console starts displaying the messages that match the message profiles available to you. You must have at least one message profile enabled to view any message. The section, *Using Message Profiles to Select the Messages to Be Monitored*, on page 21-10 describes how to enable and disable your message profiles.

Note

If console message consolidation is disabled, you can monitor local messages only. (Message consolidation is enabled or disabled in the CCONSOLIDATN parameter group. See the *Automation Services Administrator Guide* for information about parameter groups.)

Note

You can also use the Command Entry facility as a consolidated console. To access the Command Entry panel, type **CMD** at a Command ===> or Select Option ===> prompt, or press F5 from your message monitor. Enter **CCON** to turn on console message consolidation. The Command Entry facility, as opposed to the message monitor, keeps the messages that scroll off the panel. That is, you can bring those messages back onto the panel by pressing the F7 or F8 scroll function keys.

The *Management Services User's Guide* provides a more detailed description of the Command Entry facility.

If the Console Is Not Displaying System Messages...

If the console is not displaying system messages, use the following procedure to investigate the cause and correct the problem. You might not need to complete all steps if the problem is corrected before the end of the procedure.

Step 1. Enter **PROFILE CC** and ensure that at least one of your message profiles is enabled.

Note

If a defined message profile is not accessible, check its status. When you load the profiles, only those with an ACTIVE status are loaded. For information about message profiles, see Chapter 20, *Defining and Maintaining Message Profiles for the Consolidated Console*.

Step 2. Enter **PROFILE** to display your console profile. Ensure that the values of the following profile parameters are as indicated:

Parameter	Value
UNSOL	YES
AOMMSG	YES
AOMMSGLV	Other than NONE

You can correct the value by issuing the following command for each relevant parameter:

`PROFILE` *`profile-parameter=parameter-value`*

The changed value is valid for the current session only. If you want to change a value permanently, change it in your user profile.

If the AOMMSG and AOMMSGLV parameters are not displayed or if the AOMMSGLV parameters cannot be changed, you need to update your user ID definition according to the guidelines in the next step; otherwise, proceed to Step 4.

Note

If you are not authorized to correct errors found in the following steps, report the errors to the administrator.

Step 3. Enter the **/UAMS.B** path to browse your user ID definition. Ensure that your AOM Details panel (page 11 of the user ID definition) displays the following values:

Field	Value
AOM Message Receipt	Y
Console Routing Codes	ALL
Message Level Screening	ALL

When these values are correct, you can then update the corresponding profile parameters as indicated in the previous step.

You should also ensure that the Initial OCS Command field on your OCS Details panel (page 6 of the user ID definition) has the value $RMCCOCS. This command ensures that the message monitor is always presented to you as a consolidated console.

Step 4. Ensure that console consolidation is activated by the CCONSOLIDATN region parameter group. See the *Automation Services Administrator Guide* for information about parameter groups.

Using Message Profiles to Select the Messages to Be Monitored

In a consolidated console, you can use predefined message profiles to select the messages you want to monitor. To access your list of message profiles, issue the **PROFILE CC** command. The Private Message Profile Control panel appears. Figure 21-2 is an example.

Figure 21-2. Private Message Profile Control Panel

```
SOLVPROD-------- ServiceView : Private Message Profile Control -----------CC ON
Command ===>                                                   Scroll ===> PAGE

                                                             E=Enable D=Disable
    ProfileName   Id   Status    Description
    CICSMESSAGES  126  ENABLED   Messages for all Production CICS resources
    ROUTCODE21    21   ENABLED   Hasp Messages
    TAPEMOUNTS    127  DISABLED  Tape Mounts for Eastern Production jobs
    **END**



 F1=Help      F2=Split     F3=Exit      F4=Return    F5=Find      F6=Refresh
 F7=Backward  F8=Forward   F9=Swap     F10=MsgFlow  F11=LstSort
```

The panel displays the list of message profiles that you can use to profile your consolidated console.

The initial status of the message profiles are as follows:

- If you disabled the message profile in your user profiles, the profile appears with a status of DISABLED.
- If you enabled the message profile in your user profiles, the profile appears with a status of ENABLED or PENDING.

Use the **D** or **E** action codes to disable or enable selected profiles for this session with your consolidated console. Enabled profiles have a status of PENDING if your monitoring environment cannot receive the requested messages (for example, if the UNSOL profile parameter has a value of NO indicating that you cannot receive unsolicited messages).

You can use the F10 (MsgFlow) function key to switch the value of the AOMMSG profile parameter between NO and YES. This parameter indicates whether you can receive AOM messages. The value must be YES for you to receive messages at your consolidated console.

Use the F11 (LstSort) function key to sort the list of message profiles by name or by ID. The initial sort is by name.

Replying to a WTOR Message From the Consolidated Console

Note

You can reply to resource or service related WTOR messages from the status or graphical monitor by using the W command.

Use the following command to reply to a local WTOR message:

```
SYSCMD REPLY wtor-id,reply-text
```

Use the following command to reply to a remote WTOR message:

```
ROUTE DOMAIN=domain-id SYSCMD REPLY wtor-id,reply-text
```

The value of *domain-id* is the domain ID of the region that sends the remote WTOR message. The ID appears as a prefix to the message if the value of your PREFSYS profile parameter is YES.

The *Command Reference* manual describes the SYSCMD and ROUTE commands.

Tip

You can use the EQUATE command to reduce the typing required when issuing a command. For example, you can equate text as follows:

```
EQUATE / SYSCMD REPLY+
EQUATE domain-id ROUTE DOMAIN=domain-id SYSCMD REPLY+
```

You can then use the following commands to reply respectively to a local or a remote WTOR message:

```
/ wtor-id,reply-text
domain-id wtor-id,reply-text
```

The *Command Reference* manual describes the EQUATE command.

To ensure that the required text strings are always equated in the region, specify the EQUATE commands in the EQUATES parameter group.

Leaving the Consolidated Console

Exit your consolidated console in one of the following ways:

- To exit the consolidated console and remain at your message monitor or your Command Entry panel, issue the **CCOFF** command. You can use the CCON command to return to the consolidated console.
- To exit the consolidated console and return to the previous panel, press F3.

22

Managing Alerts

The alert monitor displays alerts that warn you of problems or impending problems in systems operations. Alerts might also be used to bring your attention to significant events (for example, reminding you to perform a task that cannot be automated). The alert monitor, however, is not meant to be a replacement of the status monitor. You should use the status monitor as the main facility to manage defined resources.

This chapter contains the following topics:

- Accessing the Alert Monitor
- Working With Alerts
- Displaying Alert History

Accessing the Alert Monitor

The alert monitor provides an integrated, correlated event notification system that indicates to operators that a problem has been detected and that some action needs to be taken. Such alerts are known as active alerts. Alerts of the current date that are active for the current region startup are displayed. You can, however, access the other alerts in the history log (see the section, *Displaying Alert History*, on page 22-6).

In a multisystem environment, you can monitor active alerts from all linked regions in a focal point region. You can monitor only local active alerts in a subordinate region.

To access the alert monitor, enter the **/ALERTS** shortcut. The monitor displays all active alerts that require operator attention.

Figure 22-1 shows an example of the alert monitor screens.

Figure 22-1. Alert Monitor Panels

```
SOLVPROD  (10.37.30)------ Alert Monitor : Alerts -----------------------------
Command ===>                                                   Scroll ===> CSR

           S/B=Browse T=Track N=Notes A=Analyze TT=TroubleTicket C=Close ?=More
    Time     Description                                Resource          Track
    09.33.58 STC SOLVDDDD on system SOLVPROD-0001 re SOLVDDDD
    09.33.57 STC CICSD on system SOLVPROD-0001 requi CICSD
    09.33.57 STC SOLVFFFF on system SOLVPROD-0001 re SOLVFFFF
    09.33.57 STC SOLVCCCC on system SOLVPROD-0001 re SOLVCCCC
    09.33.57 STC SOLVBBBB on system SOLVPROD-0001 re SOLVBBBB
    **END**
```

To display additional information, press F11.

```
 F1=Help      F2=Split     F3=Exit      F4=History   F5=Find      F6=Hold
 F7=Backward  F8=Forward   F9=Swap                   F11=Right
```

```
SOLVPROD  (10.40.08)------ Alert Monitor : Alerts -----------------------------
Command ===>                                                   Scroll ===> CSR

           S/B=Browse T=Track N=Notes A=Analyze TT=TroubleTicket C=Close ?=More
    Time     Description                                            S System
    09.33.58 STC SOLVDDDD on system SOLVPROD-0001 requires operator  4 SOLVPROD
    09.33.57 STC CICSD on system SOLVPROD-0001 requires operator att 4 SOLVPROD
    09.33.57 STC SOLVFFFF on system SOLVPROD-0001 requires operator  4 SOLVPROD
    09.33.57 STC SOLVCCCC on system SOLVPROD-0001 requires operator  4 SOLVPROD
    09.33.57 STC SOLVBBBB on system SOLVPROD-0001 requires operator  4 SOLVPROD
    **END**
```

The monitor refreshes your screen each time an alert arrives. The clock in the title line indicates when the screen was refreshed last. If alert activities are causing interruptions to what you are doing, press F6 (Hold) to disable screen refresh. To resume display of new alerts, press F6 (Resume) again.

For an explanation of the information displayed by, and the commands available on, the alert monitor, see the online help.

Sorting Alerts

Alerts are color coded by severity. They are sorted in order of severity, then time—the most severe alerts are listed first, then, in each category of severity, the most recent of the alerts are listed first. You can change the sort order by using the SORT command. You can locate an alert by the sorted fields by using the LOCATE command.

Filtering Alerts

You can restrict the alerts displayed by using filters. Enter **FILTER** at the Command prompt, and select the required filter.

You can define alert filters from the Filter Definition List panel. To access the list, enter the **/ALADMIN.F** path or the **DEFINE FILTER** command.

Change the Display Format

Display format determines what and how information is displayed. To change the alert monitor format, enter **FORMAT** at the Command prompt, and select the required format.

You can define alert formats from the List Definition List panel. To access the list, enter the **/ALADMIN.L** path or the **DEFINE FORMAT** command.

Working With Alerts

The alert monitor displays the alert when it arrives. Alerts may be closed automatically by the region or manually by the operator. When an alert is closed, it is removed from the alert monitor.

Typically, when an alert arrives, you would proceed as follows:

Step 1. Browse the alert to find out whether any suggested recommended actions are provided (see the next section, *Viewing the Details of an Alert*, on page 22-4).

If the alert is generated by the region on behalf of a defined service or resource, you can access the status monitor to work on that service or resource (see the section, *Accessing the Status Monitor From the Alert Monitor*, on page 22-5).

Step 2. To respond to an alert, you should indicate to other users that you will be working on it.

Enter **T** beside the alert. Your user ID is displayed in the Track column.

Step 3. Perform any necessary actions to remove the alert condition.

Step 4. Record notes that provide future reference information about this alert in the alert definition (see the section, *Adding Operator Notes to an Alert*, on page 22-5).

Step 5. After the alert condition is resolved, close the alert (if required) (see the section, *Closing an Alert*, on page 22-6).

Note

Alerts can be forwarded to other systems by using the $AMEVFWD procedure (see help).

Viewing the Details of an Alert

Enter **B** beside an alert to view its details. An alert comes with the following information:

- General information such as severity level, the source of the alert, and update history
- Possible causes of the alert and any recommended actions

To print the details, enter **PRINT** at the Command ===> prompt.

Accessing the Status Monitor From the Alert Monitor

If an alert is generated by the region on behalf of a defined service or resource, you can access the status monitor to work on that service or resource.

To access the status monitor, enter **A** beside the alert. (You might need to wait for a short time before the status monitor is displayed.)

For information about how to use the status monitor, see Chapter 13, *Monitoring and Controlling Services and Resources*.

Accessing the Transient Log From the Alert Monitor

If an alert is generated by the region on behalf of a service or resource, you can access the corresponding transient log.

To access the log, enter **TL** beside the alert.

For information about how to use the transient log, see the section, *Accessing a Transient Log*, on page 13-29.

Raising a Problem Ticket for an Alert

If your system administrator has implemented the interface for raising a problem ticket, you can apply the TT action at the Alert Monitor panel.

The TT action produces a request for a problem ticket as defined to your region.

For information about how to implement the problem ticket interface, see the *Automation Services Administrator Guide*.

Adding Operator Notes to an Alert

You can add notes to an alert for future reference. For example, you might want to leave the next operator some notes about an alert that has not yet been closed. Notes are also added automatically to provide a history of the actions performed on the alerts.

To add notes to an alert, proceed as follows:

Step 1. Enter **N** beside the alert to display the Alert Notes panel.

Step 2. Press F4 (Add) to enable you to add notes.

Step 3. Enter your notes about the alert. See the online help about how to use the editor.

Step 4. When you finish entering your notes, press F3 (File) to save the notes with the alert.

Closing an Alert

Alerts may be closed automatically by the region (internally or by using the DELALERT macro) or manually.

For example, an alert might be generated because a resource entered a degraded state. When the degraded condition is corrected, the alert is closed automatically. Another alert might be a reminder alert and needs to be closed manually.

To close an alert, enter **C** beside the alert to close it.

Displaying Alert History

The alert history lists all alerts, both active and closed, that occurred during a predefined period.

To view the alert history, press F4 (History) from the alert monitor. The alerts for the current date are displayed. If you want to display the alerts for the other dates, use the DATE command. For more information about the command, see the online help.

To view the history of actions performed on an alert, enter **N** beside it.

Note

To customize how long an alert is retained in the alert history log, enter the **/ALADMIN.H** path to display the History Logging Parameters Definition panel.

Part IV

Reference Material

A

Commands

You issue commands to perform various actions, depending on your authority level. The types of commands are status monitor line and primary commands.

Note

Automation Services commands are defined in the knowledge base. To access the list of command definitions, enter the **/ASADMIN.C** path.

This appendix describes the commands registered in the region. The commands that are available depend on the product you are using.

Line Commands

You use line commands to perform actions on listed items. Each line command acts on a single selected item on the list. For example, a line command can act on a single resource. You enter a line command beside the required item. To list the available commands for an item, enter **?** beside it.

Line Command Parameters

You can issue line commands programmatically (for example, by using the SVCMD macro). The command parameters are as follows:

SYSNAME=*system-image-name*
: Names the system image that owns the target of the command. The default is the local active system image name.

VERSION=*system-image-version*
: Specifies the version of the system image. The default is the local active system image version.

Class=*class-number*
: Specifies the number that indicates the class of the resource. See Appendix H, *Knowledge Base Definition Classes*, for the list of valid class numbers.

NAME=*target-name*
: Names the target of the command.

Primary Commands

You use primary commands to perform general processing (for example, loading a system image and its resources). You enter a primary command at the Command ===> prompt. To list the commands, enter **?** at the prompt.

Primary Command Parameters

You can issue the following commands programmatically (for example, by using the SVCMD macro): GLOBAL, LOAD, SHUTFORCE, SHUTSYS, and STARTSYS. The command parameters are as follows:

SYSNAME=*system-image-name*
Names the system image that is the target of the command. The default is the local active system image name.

VERSION=*system-image-version*
Specifies the version of the system image. The default is the local active system image version.

GLOBAL Command Parameters

The GLOBAL command has the following additional parameter:

MODE={AUTOMATED | MANUAL}
Specifies the global operation mode imposed on the active system image.

LOAD Command Parameters

The LOAD command has the following additional parameters:

NEWSYS=*new-system-image-name*
Names the system image to be loaded.

NEWVERS=*new-system-image-version*
Specifies the version of the system image to be loaded.

MODE={AUTOMATED | MANUAL}
Specifies the global operation mode imposed on the loaded system image.

WARM={NO | YES}
Specifies whether the system image is to be warm loaded.

B

System Console Interface

If VTAM fails or is not active, with only the system console available for communicating with the system, you can still influence the region from the system console by using the $RMCONS NCL procedure. You can also view the transient log of a resource on your system console by using the $RMCLOG NCL procedure.

This appendix describes the $RMCONS and the $RMCLOG NCL procedures.

$RMCONS NCL Procedure

Use the $RMCONS NCL procedure to issue commands to the region from a system console. The syntax is as follows:

```
F started-task-name, $RMCONS parameter-1 … parameter-n
```

started-task-name

Names the started task that establishes the region.

parameter-1 … parameter-n

Determines the issued command. The two types of commands are line commands and primary commands.

Line commands are resource specific and are identified by the LCMD parameter.

Primary commands are *not* resource specific and are identified by the PCMD parameter.

Line Commands

You can issue line commands by using the $RMCONS NCL procedure.

The syntax is as follows:

```
$RMCONS LCMD=line-command    NAME=name
                             CLASS=class-number
                             [SYSNAME=system-image-name
                             VERSION=system-image-version]
```

LCMD=*line-command*
Specifies the line command to issue. The A, ASA, ASD, ASF, ASI, ASS, ASU, CHK, DEL, DSA, DSI, DSR, EDR, EDS, LR, MA, MI, MM, MR, MRS, RES, S, T, TF, TRI, and TRR line commands are supported.

NAME=*name*
Names the service or resource for which you want to issue the command.

CLASS=*class-number*
Identifies the class of the service (class 61) or resource. See Appendix H, *Knowledge Base Definition Classes*, for information about class numbers.

SYSNAME=*system-image-name*
Names the system image that owns the service (the $SERVICE 0001 image) or resource.

VERSION=*system-image-version*
Identifies the version of the system image.

The default system image is the image loaded on the local system.

Primary Commands

You can issue the following primary commands by using $RMCONS:

CHECKALL
Sets the actual state of all resources to UNKNOWN and then uses the display method defined for each resource to check the status of that resource. The syntax is as follows:

```
$RMCONS      PCMD=CHECKALL
```

GLOBAL

Sets the global operation mode. The syntax is as follows:

```
$RMCONS     PCMD=GLOBAL
            PARMS="MODE={AUTOMATED | MANUAL}"
```

LOAD

Loads a system image. The syntax is as follows:

```
$RMCONS     PCMD=LOAD
            PARMS="NEWSYS=new-system-image-name
                   NEWVERS=new-system-image-version
                   MODE={AUTOMATED | MANUAL}"
```

QLOAD

Retrieves information about the active local system image and miscellaneous statuses. The syntax is as follows:

```
$RMCONS     PCMD=QLOAD
```

SHUTFORCE

Forces all resources into the AUTOMATED operation mode and shuts down the resources. The syntax is as follows:

```
$RMCONS     PCMD=SHUTFORCE
```

SHUTSYS

Shuts down only the resources that are in the AUTOMATED operation mode. The syntax is as follows:

```
$RMCONS     PCMD=SHUTSYS
```

STARTSYS

Restarts the resources that are shut down. The syntax is as follows:

```
$RMCONS     PCMD=STARTSYS
```

$RMCLOG NCL Procedure

Use $RMCLOG to view the transient log of a service or resource on a system console. The syntax is as follows:

```
F started-task-name, $RMCLOG   NAME=name
                               [CLASS={02|class-number}]
                               [START=list-from-time]
                               [MESSAGES={20 | number-of-messages}]
                               [LOGDATA={TEXT | ALL}]
                               [LOG={NO | YES}]
```

started-task-name
Names the started task that establishes the region.

NAME=*name*
Names the service or resource for which you want to view the transient log.

CLASS={02 | *class-number*}
Identifies the class of the service (class 61) or resource. See Appendix H, *Knowledge Base Definition Classes*, for information about class numbers. The default is the class of started task resources.

START=*list-from-time*
Specifies the time from which to begin displaying the logged messages. The default is from the beginning of the transient log.

MESSAGES={20 | *number-of-messages*}
Specifies the number of the most recent messages to display. The default is the last 20 messages.

LOGDATA={TEXT | ALL}
Specifies whether to display message text only (TEXT) or to display message text and attributes (ALL).

LOG={NO | YES}
Specifies whether to send the displayed messages to the activity log.

C

Exit NCL Procedures

Use exit NCL procedures to perform special processing.

There are two supplied exit NCL procedures for recording problems to the following applications:

- SOLVE:Problem application
- Operations Information Management (OIM) Problem Management application

These exits enable you to record problems automatically as the problems occur.

This appendix describes the supplied exit NCL procedures.

Exit to the SOLVE:Problem Application

The $RMPB06S NCL procedure is an exit to the SOLVE:Problem application. The syntax is as follows:

```
$RMPB06S ACBNAME=acb-name
```

acb-name

Identifies the link to the region in which the SOLVE:Problem application is running. You can issue the SHOW PARMS command from the SOLVE:Problem region to determine its ACB name. The name is specified by the PRI parameter.

Return Codes

A zero return code indicates that the NCL procedure was successful. A nonzero return code indicates that the NCL procedure failed.

If an error occurs during the execution of the $RMPB06S NCL procedure, the error is logged as a severity 1 message (which appears in red in a transient log). The message contains one of the following return codes:

Return Code	Description
4	Failed execution
8	Syntax error

Consideration

Before you can use this exit, you must perform the following:

- Copy the $RMPB07S NCL procedure to the region in which the SOLVE:Problem application is running.
- Authorize the following user IDs in the SOLVE:Problem region:
  - Automation Services BSYS background user ID
  - IDs of the users who may raise problem tickets manually
- In the SOLVE:Problem region, define a link to the Automation Services region.

For more information about the implementation, see the *Automation Services Administrator Guide*.

Example

Automation Services supplies a sample process, PROBSOLV, that invokes this procedure. The process uses the ACB parameter to pass *acb-name* to the procedure.

Exit to the OIM Problem Management Application

The $RMPB01S NCL procedure is an exit to the OIM Problem Management application. The syntax is as follows:

```
$RMPB01S    ACTION=ADD
            LINKNAME=link-name
            [ASGROUP=user-group-name]
            [CAT={GDSP | symptom-category}]
            [FAULT={OTHER | fault-code}]
            [SEV={4 | severity-code}]
```

link-name

Identifies the link to the region in which the OIM software is running.

user-group-name

Names the OIM user group to which the problem is assigned.

The default is the value of the Group field on the Owner Details panel under First Level Support Details in a resource definition.

symptom-category

Identifies the component that causes the problem.

The default is GDSP, indicating system software. See the *Operations Information Management Problem Management User Guide* for the list of valid values.

fault-code

Indicates the type of problem. The length should *not* be more than 15 characters.

The default is OTHER.

severity-code

Indicates the severity of the problem.

Valid values are 1 through 4.

The default is 4. A lower value indicates a more severe problem. See the *Operations Information Management Problem Management User Guide* for a description of how to use the severity codes.

Return Codes

A zero return code indicates that the NCL procedure was successful. A nonzero return code indicates that the NCL procedure failed.

If an error occurs during the execution of the $RMPB01S NCL procedure, the error is logged as a severity 1 message (which appears in red in a transient log). The message contains one of the following return codes:

Return Code	Description
4	Failed execution
8	Syntax error

Consideration

Before you can use this exit, you must also copy this procedure to the region in which the OIM software is running.

Example

Automation Services supplies a sample process, PROBOIM, that invokes this procedure. The process uses the following parameters to pass values to the procedure:

Process Parameter	Value Passed
LINK	*link-name*
GRP	*user-group-name*
CAT	*symptom-category*
FAULT	*fault-code*
SEV	*severity-code*

D

Variables

Variables enable you to access knowledge base data, find out about the status of services and resources, and extract information about messages.

Variables are of the following types:

- Variables that contain data in the knowledge base
- Variables that contain status information
- Variables that contain message information

This appendix describes these variables. The variables that are available depend on the product you are using. Refer to the individual product user's guide for other variables that are specific to the product you are using.

Knowledge Base Variables

Use the variables in Table D-1 to retrieve data from the knowledge base. Variables that are specific to a particular product are identified in parentheses.

You can use the variables to pass values to the fields in a template, to the fields in a service or resource definition, to the parameters in macros used in the definition, and to NCL procedures invoked from the definition. Figure D-1 shows an example of the use the &ZRMDBNAME variable in the WTOR macro.

Figure D-1. Example—Using Knowledge Base Variables

```
SOLVPROD--- Automation Services : WTOR Issue MACRO Parameter Definition -------
Command ===                                                   Function=BROWSE

.- WTOR Text To Be Issued -------------------------------------------------.
|                                                                           |
| WTOR Text... &ZRMDBNAME START - REPLY WARM OR COLD                        |
|                                                                           |
`- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -'
```

Table D-1. Knowledge Base Variables

Variables	Description
&ZRMDBCLASS	Contains the class number. See Appendix H, *Knowledge Base Definition Classes*, for information about classes.
&ZRMDBCLDESCS	Contains the class name. See Appendix H, *Knowledge Base Definition Classes*, for information about classes.
&ZRMDBREL1 to &ZRMDBREL25	Contain the names of the immediate parents.
&ZRMDBSCHED	Contains the name of the availability map.
&ZRMDBSYSNAME	Contains the name of the system image.
&ZRMDBVERSION	Contains the version of the system image.

(Sheet 1 of 4)

Table D-1. Knowledge Base Variables

Variables	Description
The following fields, if applicable, are displayed on the Description panel of a service or resource definition:	
&ZRMDBACBNAME (MSP, VOS3, CICS and OS/390)	Contains the value of the ACB Name field.
&ZRMDBACTNAME	Contains the same value as &ZRMDBNAME. However, if the value is qualified, the qualifier is dropped. For example, a CICS resource name has the owner as the qualifier, *owner-name.resource-name*. This variable will return only *resource-name*.
&ZRMDBMODE	Contains the value of the Operation Mode field. The value can be AUTOMATED, IGNORED, MANUAL, or OFF.
&ZRMDBNAME	Contains the value of the field that identifies the service or resource.
&ZRMDBOWNRCL#	Contains the class number of the resource owner.
&ZRMDBOWNRCLS	Contains the value of the field that identifies the class name of the resource owner.
&ZRMDBOWNRNME	Contains the value of the field that identifies the owner of the resource.
&ZRMDBREMOCL# (CICS)	Contains the class number of the remote owner of the link resource.
&ZRMDBREMOCLS (CICS)	Contains the value of the field that identifies the class name of the remote owner of the link resource.
&ZRMDBREMONME (CICS)	Contains the value of the Remote CICS Region Name field.
&ZRMDBREMOSMF (CICS)	Contains the value of the SMFID field.
&ZRMDBSDESC	Contains the value of the Short Description field.
&ZRMDBTEMPLAT (MSP, VOS3, CICS and OS/390)	Contains the value of the TemplateName field.
&ZRMDBTYPE	Contains the value of the Type field.

(Sheet 2 of 4)

Table D-1. Knowledge Base Variables

Variables	Description
The following field is displayed on the Activation Details panel of a resource definition:	
&ZRMDBINITCMD	Contains the value of the Command field.
	The following field is displayed on the Inactivation Details panel of a resource definition:
&ZRMDBTMRTCMD	Contains the value of the Command field.
	The following field is displayed on the Force Inactivation Details panel of a resource definition:
&ZRMDBFTRMCMD	Contains the value of the Command field.
	The following field is displayed on the Display and Heartbeat Details panel of a resource definition:
&ZRMDBDISPCMD	Contains the value of the Command field.
The following fields are displayed on the Automation Log Details panel of a service or resource definition:	
&ZRMDBLOGAUDT	Contains the value of the Log Internal Audit Trail field.
&ZRMDBLOGMVSC	Contains the value of the Log to Console field.
&ZRMDBLOGOCS	Contains the value of the Log to OCS Window field.
&ZRMDBLOGRAM	Contains the value of the Log to Automation Log field.
&ZRMDBLOGSYS	Contains the value of the Log All System Msgs field.
&ZRMDBLOGTSIZ	Contains the value of the Log Table Size field.

(Sheet 3 of 4)

Table D-1. Knowledge Base Variables

Variables	Description
The following fields are displayed under First Level Support Details on the Owner Details panel of a service or resource definition:	
&ZRMDBREOGRP1	Contains the value of the Group field.
&ZRMDBREOPAG1	Contains the value of the Pager Number field.
&ZRMDBREOPHA1	Contains the value of the Phone After Hours field.
&ZRMDBREOPHB1	Contains the value of the Phone Business Hours field.
&ZRMDBREOWNN1	Contains the value of the Name field.
&ZRMDBRESOID1	Contains the value of the Userid field.
The following fields are displayed under Second Level Support Details on the Owner Details panel of a service or resource definition:	
&ZRMDBREOGRP2	Contains the value of the Group field.
&ZRMDBREOPAG2	Contains the value of the Pager Number field.
&ZRMDBREOPHA2	Contains the value of the Phone After Hours field.
&ZRMDBREOPHB2	Contains the value of the Phone Business Hours field.
&ZRMDBREOWNN2	Contains the value of the Name field.
&ZRMDBRESOID2	Contains the value of the Userid field.

(Sheet 4 of 4)

Status Variables

Use the variables in Table D-2 to retrieve the status of a service or resource.

You can use the variables in the same places where you can use knowledge base variables.

Table D-2. Status Variables

Variables	Description
&ZRMSTACTSTAT	Contains the base actual state only. The &ZRMSTPHYSTAT can contain any of the detectable actual states. The section, *Using the RECOVERED Actual State*, on page 5-26 provides an example of the difference between the values contained in these two variables. The value can be ACTIVE, INACTIVE, STARTING, STOPPING, or UNKNOWN.
&ZRMSTACTDESC	Contains information about the actual state in the &ZRMSTPHYSTAT variable (for example, the message that caused the state to be set). The information is displayed on the Modes and States panel when an S (Status) command is executed.
&ZRMSTACTSTA0	Contains information about the base actual state in the &ZRMSTACTSTAT variable (for example, the message that caused the state to be set).
&ZRMSTARMFLAG	Contains information that indicates whether a resource is registered to the automatic restart manager (ARM) or if ARM is restarting the resource. The value can be YES, NO, or RECOVERY.
&ZRMSTAUTSTAT	Contains the automation status. The value can be FAILED, IN PROGRESS, or NONE. The information is displayed on the Modes and States panel when an S (Status) command is executed.
&ZRMSTCLASS	Contains the class number (for example, 17 for the user class). See Appendix H, *Knowledge Base Definition Classes*, for information about classes.
&ZRMSTCLDESCL	Contains the class description.
&ZRMSTCLSNAME	Contains the class name. See Appendix H, *Knowledge Base Definition Classes*, for information about classes.
&ZRMSTSDESC	Contains the short description of the resource as specified in the Short Description field of the resource definition.

(Sheet 1 of 3)

Table D-2. Status Variables

Variables	Description
&ZRMSTDESSTAT	Contains the desired state scheduled by the availability map. The value can be ACTIVE, INACTIVE, or RESET.
&ZRMSTDESSTA0	Contains information about the scheduled desired state (for example, the time when the state is set).
&ZRMSTDSTACUR	Contains the current desired state. The value can be ACTIVE or INACTIVE.
&ZRMSTDSTACU0	Contains information about the current desired state (for example, whether the state is scheduled or overridden).
&ZRMSTDSTAOV	Contains the desired state override. The value can be ACTIVE, INACTIVE, or NONE.
&ZRMSTDYNAMIC	Contains the value that indicates whether the definition for the resource is dynamic. The value can be NO or YES.
&ZRMSTEXIT	Contains the value that indicates whether a state change exit process exists. The value can be NO or YES.
&ZRMSTEXTDISP	Contains any activated extended display.
&ZRMSTMODECUR	Contains the current operation mode. The value can be AUTOMATED, IGNORED, or MANUAL.
&ZRMSTMODECU0	Contains information about the current operation mode (for example, whether the mode is overridden).
&ZRMSTMODEDB	Contains the operation mode specified in the definition. The value can be AUTOMATED, IGNORED, or MANUAL.
&ZRMSTMODEMAP	Contains the operation mode scheduled by the availability map. The value can be AUTOMATED, IGNORED, MANUAL, NONE, or RESET.
&ZRMSTMODEOV	Contains the operation mode override. The value can be AUTOMATED, IGNORED, MANUAL, or NONE.
&ZRMSTNAME	Contains the name of the resource or service.
&ZRMSTNMDID	Contains the domain ID of the region that monitors and controls the resource.

(Sheet 2 of 3)

Table D-2. Status Variables

Variables	Description
&ZRMSTNRMSTAT	Contains the logical state. The value can be ATTENTION, DEGRADED, FAILED, INERROR, OK, PENDING, STARTING, STOPPING, or UNKNOWN.
&ZRMSTOBJID	Contains the object ID of the record. The object ID is the concatenation of system image name, system image version, class number, and name.
&ZRMSTPHYSTAT	Contains the actual state. The value can be ACTIVE, DEGRADED, FAILED, INACTIVE, STARTING, STOPPING, or UNKNOWN.
&ZRMSTSYSNAME	Contains the name of the system image to which the resource belongs.
&ZRMSTTYPE	Contains the resource type as specified in the Type field of the resource definition.

(Sheet 3 of 3)

Active System Image Variables

The following variables return the name and version of the local active system image: &ZRMSYSNAME and &ZRMVERSION.

Message Variables

Use the variables in Table D-3 to retrieve information about a message. Some of the variables have equivalent AOM NCL system variables that are described in more detail in the *Network Control Language Reference*, Volume 2.

You can use the variables to pass values to a message rule, to the parameters of macros used in the rule, and to NCL procedures invoked from the rule. Figure D-2 shows an example of the use the &ZMSGWORD*n* variables in the text to be displayed in response to a tape mounting request being satisfied.

Figure D-2. Example—Using Message Variables

```
SOLVPROD-- ResourceView : TAPE 380 Define Extended Display Attribute -SOLV-0001
Command ===>                                                   Function=UPDATE

. Extended Display (EXTDISP) ------------------------------------------------.
| Mount performed Volser=&ZMSGWORD5 Job=&ZMSGWORD8                            |
'-----------------------------------------------------------------------------'
```

Caution

The special messages prefixes ($AA-, $DN-, or $MN-) may be used in ResourceView to enhance functionality. A message captured by using such a prefix has the prefix included in the message variables.

Table D-3. Message Variables

Variables	Description
&ZMSGATEXT (&AOMATEXT)	Contains the text of the current line of a message.
&ZMSGAWORD*n*	Contains a word in the current line of a message. *n* is a number indicating the position of the word in the message and is in the range 1 through 32. Words are delimited by a blank or a comma.
&ZMSGJOBID (&AOMJOBID)	Contains the identification number of the job that sends the message.
&ZMSGJOBNM (&AOMJOBNM)	Contains the name of the job that issued the message. The value may be overridden by a job name supplied by JES. Thus, messages originating from JES can have the name of the target job, rather than the JES job name.
&ZMSGMSGID (&AOMMSGID)	Contains the first word of the message text and is equivalent to &ZMSGWORD1. If the message is a multiline WTO or WTOR message, &ZMSGMSGID contains the first word of the major line text.

(Sheet 1 of 3)

Table D-3. Message Variables

Variables	Description
&ZMSGTEXT (&AOMTEXT)	Contains the text of the message. If the message is a multiline WTO or WTOR message, &ZMSGTEXT contains the first (major) line of the message. Use the &ZMSGATEXT variable to retrieve the current line (which can be a minor line) of a message.
&ZMSGTIME	Contains the time of message arrival, in the form *hh.mm.ss*. The value can be in the range 00.00.00 through 23.59.59.
&ZMSGWORD*n*	Contains a word in the message text. If the message is a multiline WTO or WTOR message, &ZMSGWORD*n* contains a word in the first (major) line of the message. *n* is a number indicating the position of the word in the line and is in the range 1 through 64. Words are delimited by a blank or a comma. Use the &ZMSGAWORD*n* variable to retrieve a word in the current line (which can be a minor line) of a message.
&ZMSGWRID (&AOMWRID)	Contains the WTOR reply number. The value is blank if the message is *not* of the WTOR type.
The following variables are available to EventView message rules only:	
&ZMSGIJOBN (&AOMIJOBN)	Contains the name of the job that issued the WTO or WTOR message.
&ZMSGSYSNM	Contains the name of the system from which the message originates. See also the EventView extended message filter criterion, *System Name*, on page 17-4.
The following variables are available to completed (or triggered) EventView message group rules:	
&ZMSGCORR	Contains the correlation key used for the triggered message group rule.
&ZMSGGRP	Contains the name of the triggered message group rule.
&ZMSGTXTN	Contains the number of messages that triggered the message group rule.
&ZMSGTXT*n*	Contains the text of one of the messages that triggered the message group rule. *n* is a message arrival sequence number and is in the range 1 to &ZMSGTXTN.

(Sheet 2 of 3)

Table D-3. Message Variables

Variables	Description
The following variables are *not* available to EventView correlation keys, replacement text, and ZREV variables:	
&ZMSGDATE	Contains the date of message arrival, in the form *dd-mmm-yyyy* (for example, 14-JUN-2000).
&ZMSGDAY	Contains the day of message arrival, in the form *ddd* (for example, WED).

(Sheet 3 of 3)

E

Macros

Macros are NCL procedures that are the building blocks of processes.

This appendix describes the supplied macros.

Supplied Macros

Table E-1 summarizes the supplied macros. A full description of each supplied macro follows this table.

Table E-1. Macros Used to Define Processes

Name	Function
AOMALERT	Generates or simulates an AOM event.
CHAIN	Links the current process to another process that continues the execution.
COMMAND	Issues a Management Services command and interprets the results.
COMPARE	Compares two values.
CONCAT	Concatenates specified data and stored the result in a variable.
DELALERT	Deletes an alert.
EVENT	Issues an event message.
EVVARGET	Retrieves the values of user-defined EventView variables.
EVVARSET	Sets the values of user-defined EventView variables.
EXECNCL	Executes an NCL procedure in line with the process.
EXTRACT	Extracts a segment from a text string.
GENALERT	Generates an alert.
GETSTAT	Discovers the status of a resource or service.
GOTO	Goes to the next process step to run.
PARSE	Parses a text string into variables.
REPLY	Replies to an outstanding write-to-operator with reply (WTOR) message.
RUNPRCSS	Runs another process in line with the current process.
SETRC	Sets the process return code.
SETSTATE	Sets the actual state and the extended display attributes of a resource or service.

(Sheet 1 of 2)

Table E-1. Macros Used to Define Processes

Name	Function
SETVARS	Creates variables for a process.
SMFWRITE	Writes a user-defined SMF record.
SNMPTRAP	Generates an SNMP trap.
STARTNCL	Starts an NCL procedure.
STOP	Stops the process after setting a return code.
SUBJOB	Submits a JCL batch job.
SUBPRCSS	Submits another process to a region and runs the process separately from the current process.
SVAPI	Executes an API call.
SVCMD	Issues a registered Automation Services command.
SYSCMD	Issues a system command and interprets the results.
TRANS	Changes all occurrences of a set of characters into another set of characters.
WAIT	Instructs a process to wait for a specified period.
WAITEVNT	Waits for an N00102 event message.
WAITSTAT	Waits for a specific resource or service state change and retrieves the status.
WTO	Sends a WTO message.
WTOR	Sends a WTOR message and waits for the reply.

(Sheet 2 of 2)

Products are shipped with the above default set of macros. The macros are designed to simplify your operations management and control tasks. (See Chapter 8, *Defining Processes to Implement Complex Operations*, for information about using Macros to define processes.)

Each macro has a set of parameters, some of which have default settings which can be modified. Other parameters do not have a default setting and must be specified by you. You specify the parameters on the Macro Parameter Definition panel. To display the Macro Parameter Definition panel, enter **P** beside the process step.

To determine which parameters (fields) on the Macro Parameter Definition screen are mandatory, press ENTER. The mandatory fields are highlighted.

Note

Variables returned by a macro can be passed as parameters to the other macros in the same process.

All macros set the return code in the &$RMMCRC variable. Unless specifically mentioned in the following descriptions, a zero return code means a successful operation and a nonzero return code means an unsuccessful operation.

AOMALERT

Where possible, avoid using this macro. If you want to generate a WTO message, use the WTO macro.

The macro is based on the &AOMALERT NCL verb. For more information about the verb, see the *Management Services Network Control Language Reference*.

CHAIN

Purpose

Links the current process to another process that continues the execution.

You can chain processes together so that process A passes control to process B if certain conditions are met. You can define processes as modules and chain them as required. Use the CHAIN macro to chain processes. The macro sets the name of the target process and any associated parameters, and passes control to the target process.

Parameters

The parameters for the CHAIN macro are set in the following fields:

Process

Is a mandatory field that specifies the name of the next process to be executed, provided that certain conditions are met. If no conditions are specified in the step containing the macro, the process executes automatically.

Parameters

Is only relevant if the process specified in the Process field contains variables that require you to pass values to them.

If the specified process does not require any passed values, this field remains empty.

Examples

In the following example, the PROC2 process is called and will be executed. PROC2 contains the variable &NAME which is given a value of TYOUNG.

Figure E-1. CHAIN Macro Parameter Definition (Example A)

```
SOLVPROD---- Automation Services : CHAIN Macro Parameter Definition -----------
Command ===>                                                   Function=UPDATE

.- Process Details -------------------------------------------------------------.
|                                                                               |
|  Process ..... PROC2                                                          |
|                                                                               |
|  Parameters .. NAME=TYOUNG                                                    |
|                                                                               |
|                                                                               |
'-------------------------------------------------------------------------------'
```

In the following example, the PROC5 process is called and will be executed. PROC5 contains the variable &NAME which is given a value of &V1. The value of the &V1 variable can be set, for example, by an EventView message action rule.

Figure E-2. CHAIN Macro Parameter Definition (Example B)

```
SOLVPROD---- Automation Services : CHAIN Macro Parameter Definition -----------
Command ===>                                                   Function=UPDATE

.- Process Details -------------------------------------------------------------.
|                                                                               |
|  Process ..... PROC5                                                          |
|                                                                               |
|  Parameters .. NAME=&V1                                                       |
|                                                                               |
|                                                                               |
'-------------------------------------------------------------------------------'
```

COMMAND

Purpose

Issues a Management Services (MS) command and interprets the results.

The COMMAND macro enables you to specify message rules to detect the required response message.

Parameters

The parameters for the COMMAND macro are set in the following fields:

Command

Is a mandatory field that specifies the name of the command to be issued by the macro.

This field can contain up to 61 characters.

Wait Time

Specifies the maximum time, in seconds, that the region waits for a response message. The time is reset each time a response is received. The macro waits until a message that matches a rule in the Message Text fields is received or until the time expires without receiving any more responses.

This field must contain a number from 1 through 9999. The default is 30.

Wait Time Expiry Return Code

Sets the return code if the expected message does not arrive before the expiry of the time specified in the Wait Time field.

This field must contain a number from 1 through 999. The default is 69.

Response Message Analysis

The following fields specify the rules that apply to response messages. You can specify up to five rules. The action codes enable you to refine the message rule definition criteria. More information on the use of action codes is available in the online help.

Message Text

Specifies the initial part of the expected response message. It can contain up to 45 characters.

Return Code

Specifies the value returned by the macro when it receives the response message. The return code must be a number from 0 through 9999.

Extended Filter

Indicates whether an extended filter has been defined. This field is empty until the Message Text and the Return Code fields are completed. If you want to define an extended filter, enter **S** beside the rule to access the Extended Message Filtering panel.

After this panel is completed, the Extended Filter field is automatically updated.

Appendix F, *Message Text Analysis*, provides more information about how to define extended message filters.

Returned Variables

Variables	Description
&$RMMCMSGTEXT	Contain the full text of the message that satisfies the message rules.
&$RMMCWORD*n*	Contain the words in the message, where *n* indicates the position of the word in the message.
&ZRMMSG*n*	Contain any messages received before the required response message, where *n* is a number indicating the order in which those messages are received.

For example, if the response to a command consists of three messages A, B, and C, and message B satisfies the message rules, then:

- &ZRMMSG1 contains message A
- &$RMMCMSGTEXT contains message B
- The macro ends on receipt of message B, and does not see message C

Example

In the following example, the COMMAND macro executes the SHOW ALLOC=RAMLOG01 command and waits for a response message. In this case, that message is `N15115 RAMLOG01 ALLOCATED SOLVBSYS` *`hh.mm...`*

If this message is not received within 30 seconds, the return code is set to 69.

An extended filter checks for the value of *hh.mm*. Detailed information about the completion of the Extended Message Filtering Panel is available from the online help.

Figure E-3. COMMAND Macro Parameter Definition

```
SOLVPROD--- Automation Services : COMMAND Macro Parameter Definition ----------
Command ===>                                                   Function=UPDATE

.- Command Details -------------------------------------------------------------.
|                                                                               |
| Command ..... SHOW ALLOC=RAMLOG01                                             |
|                                                                               |
| Wait Time ... 30   Wait Time Expiry Return Code ... 69                        |
'-------------------------------------------------------------------------------'
.- Response Message Analysis ---------------------------------------------------.
|                                  D=Delete Extended Filter  S=Extended Filter |
|                                                                               |
|           Message Text                                      Return  Extended |
|                                                              Code   Filter?  |
|   ___     N15115 RAMLOG01 ALLOCATED SOLVBSYS                 0       YES    |
```

Figure E-4. Extended Message Filtering Panel

```
SOLVPROD------- Automation Services : Extended Message Filtering --------------
Command ===>                                                   Function=UPDATE

. Extended Filter Definition ---------------------------------------------------.
|                                                                               |
| Message Text .......... N15115 RAMLOG01 ALLOCATED SOLVBSYS                    |
|                                                                               |
| Wildcard Character .... *                                                     |
'-------------------------------------------------------------------------------'
. Message Text Analysis --------------------------------------------------------.
|                                                                               |
|   Strt Word      Scan                                                         |
|   Pos  Num  Opr  Text                                                         |
| 1 001  005  LT   07.00                                                        |
| 2 ___  ___  ___  ____________________________________________________________ |
| 3 ___  ___  ___  ____________________________________________________________ |
| 4 ___  ___  ___  ____________________________________________________________ |
| 5 ___  ___  ___  ____________________________________________________________ |
|                                                                               |
| Expression e.g (1 and (2 or 3)) ______________________________________________ |
'-------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=OK
                           F9=Swap                                F12=Cancel
```

COMPARE

Purpose

Compares two values. You can:

- Compare the value in a variable with the value in another variable
- Compare the value in a variable with a fixed value
- Select a subset of the value to be compared

Parameters

The parameters for the COMPARE macro are set in the following fields:

Variable Name 1

Is a mandatory field that is used to specify the variable that contains the value you want to compare. You can use the Start of Compare and the Length fields to compare a substring of the value.

Operator

Is a mandatory field that specifies the relational operator to use for the comparison. The value must be one of the following:

- CT (contains)
- EQ (is equal to)
- NE (is not equal to)
- GT (is greater than)
- GE (is greater than or equal to)
- LT (is less than)
- LE (is less than or equal to)

Variable Name 2

Is used to specify a variable that contains the value you want to use as the basis of the comparison.

Constant Value

Is used when you want to specify a fixed text string as the basis of the comparison.

Note

To use a null value for comparison, leave both the Variable Name 2 field and the Constant Value field blank.

Start of Compare

Indicates the start position of the string in variable 1 from which the comparison takes place. The value of this field must be a number from 1 through 255.

Length

Indicates how much of the string in variable 1 is to be compared. The value, if specified, must be a number from 1 through 255. If a value is not specified, the full string or substring, starting from the specified start position, is compared.

Return Codes

Return Code	Description
0	Comparison criteria satisfied
4	Comparison criteria not satisfied

Example

In the following example, the COMPARE macro checks whether the value of the &A3 variable is greater than the value of the &A2 variable.

Figure E-5. COMPARE Macro Parameter Definition

```
SOLVPROD--- Automation Services : COMPARE MACRO Parameter Definition ----------
Command ===>                                                   Function=BROWSE

.- COMPARE Parameters -------------------------------------------------------.
|                                                                            |
|  Syntax .......... (Variable 1) OPERATOR (Variable 2)> or (Constant)       |
|                                                                            |
|                                                                            |
|  Variable Name 1 ....... A3            (Enter a valid variable name)       |
|                                                                            |
|  Operator .............. GT            (EQ,NE,GT,GE,LT,LE,CT)              |
|                                                                            |
|  Variable Name 2 ....... A2            (Enter a valid variable name)       |
|                                                                            |
|    or Constant Value ...                                                   |
|                                                                            |
|  Start of Compare ...... 1             (Start position within variable 1)  |
|                                                                            |
|  Length ...............                (Enter length of comparison)        |
|                                                                            |
'----------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

CONCAT

Purpose

Concatenates specified data and stores the result in a variable.

Parameters

The parameters for the CONCAT macro are set in the following fields:

Data

Specifies the data that you want to concatenate. The data can contain text and variables separated by spaces, which may either be preserved or be removed during concatenation depending on the value in the Preserve Blanks field. The result cannot be more than 256 characters. Excess characters are truncated.

Preserve Blanks

Specifies whether the spaces entered in the Data field should be preserved or removed during concatenation.

Variable Name

Specifies the name of the variable that contains the result of the concatenation.

Example

In the following example, the CONCAT macro creates the &SRCHKEY variable, which contains the value 55LOCN*x*, where *x* is the value of the &LOCATION variable:

Figure E-6. CONCAT Macro Parameter Definition

```
SOLVPROD--- Automation Services : CONCAT Macro Parameter Definition -----------
Command ===>                                                     Function=BROWSE

.- Data to be Concatenated -----------------------------------------------------.
|                                                                                |
|  Enter data to be concatenated below. Data can contain variables and/or        |
|  text separated by blanks.                                                     |
|                                                                                |
|  Data ....... 55 LOCN &LOCATION                                                |
|                                                                                |
|                                                                                |
|  Preserve blanks ... NO   (YES or NO)                                          |
|                                                                                |
|  Notes: Blanks within variables are always preserved.                          |
|         The created variable has a maximum length of 256 characters.           |
|         Excess characters will be truncated and a warning message logged.      |
'--------------------------------------------------------------------------------'
.- Variable to be Created ------------------------------------------------------.
|                                                                                |
|  Variable Name ..... SRCHKEY                                                   |
'--------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

DELALERT

Purpose

Deletes a user-generated alert.

Parameters

The DELALERT macro has only one parameter: the identifier of the alert to be deleted. The identifier is the alert reference key with which the alert was generated.

Example

In the following example, the DELALERT macro deletes an alert called UPSALERT9001:

Figure E-7. DELALERT Macro Parameter Definition

```
SOLVPROD--------------- EventView : Alert Delete -----------------------------
Command ===>                                                   Function=BROWSE

. Alert Reference Key -------------------------------------------------------.
|                                                                            |
| Reference ... UPSALERT9001                                                 |
```

See Also

The section, *GENALERT*, on page E-27.

EVENT

Purpose

Issues an N00102 event message that can be retrieved by NCL processes. To receive the message, the NCL process must be profiled for the event by using the PROFILE EDS command. See the *Command Reference* for a description of the command.

You can wait for the event message from within a process by using the WAITEVNT macro.

Parameters

The parameters for the EVENT macro are set in the following fields:

Name

Identifies the event. The name can contain alphanumeric, @ and # characters, and . or _ characters (for example, SESSION_COMPLETION).

Type

Indicates the type of information contained in the event. Use the types as follows:

APPLICATION	To contain user-defined information
ACCESS	To contain security alarm information
CONFIGURATION	To contain object definition and relationship information
PROCEDURAL	To contain scheduling and process control information
SERVICEABILITY	To contain availability, degradation, error, fault, and recovery information
UTILISATION	To contain accounting, performance, response time, and statistics information

Scope

Specifies the scope of delivery of the event. Valid values are as follows:

REGION	Restricts event delivery to processes within the region of the event issuer.
SYSTEM	Enables event delivery to processes within the entire region.

Object

Specifies a user-defined object class for the resource or resource pair for which the event is issued. If the value contains embedded spaces, the spaces are translated to underscore (_) characters.

Resource

Names the resource or resource pair (separated by a comma) for which the event is issued. An event can be issued for a pair of resources (for example, a session pair).

Reference

Specifies a code for the event.

Data

Specifies the data in the event.

Example

In the following example, the EVENT macro issues an OPERATIONS_CONDITION event. The event contains operations information in the &ZREVATTENDED user-defined EventView variable.

Figure E-8. EVENT Macro Parameter Definition

```
SOLVPROD---- Automation Services : EVENT Macro Parameter Definition -----------
Command ===>                                                     Function=BROWSE

.- Event Parameters -------------------------------------------------------.
|                                                                           |
| Name........ OPERATIONS_CONDITION                                         |
|                                                                           |
| Type.......+ SERVICEABILITY                                               |
| Scope......+ SYSTEM                                                       |
|                                                                           |
| Object......                                                              |
| (Blanks will be translated to underscores)                                |
|                                                                           |
| Resource....                                                              |
| Reference...                                                              |
|                                                                           |
| Data..... &ZREVATTENDED                                                   |
|                                                                           |
|                                                                           |
|                                                                           |
'---------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

See Also

The section, *WAITEVNT*, on page E-66.

EVVARGET

Purpose

Retrieves the values of user-defined EventView variables. Once obtained, these variables can be used in subsequent steps in the process that contains the macro.

Parameters

The EVVARGET macro has only one set of parameters: the character strings that, together with the ZREV prefix, form the names of the variables from which you want to retrieve the values.

Example

In the following example, the EVVARGET macro retrieves the value of the &ZREVATTENDED variable. A message can then be sent describing the condition indicated by the value. The message can be used to modify operations behavior.

Figure E-9. EVVARGET Macro Parameter Definition

```
SOLVPROD--- Automation Services : EVVARGET Macro Parameter Definition ---------
Command ===>                                                   Function=BROWSE

. Get EventView Variables -------------------------------------------------.
|                                                                          |
|      Name                                                                |
| ZREV ATTENDED                                                            |
| ZREV                                                                     |
| ZREV                                                                     |
| ZREV                                                                     |
| ZREV                                                                     |
| ZREV                                                                     |
|                                                                          |
'--------------------------------------------------------------------------'
```

See Also

The section, *EVVARSET*, on page E-20 and the section, *Using EventView Variables*, on page 16-12.

EVVARSET

Purpose

Sets the values of user-defined EventView variables.

Parameters

The parameters for the EVVARSET macro are set in the following fields:

Name
Specifies the character string that, together with the ZREV prefix, forms the name of the variable.

Value
Specifies the value of a named variable.

Example

You want the EventView rules to react to messages differently depending on whether the system is attended, as indicated by the value of the &ZREVATTENDED variable. Use the WTOR macro to send a WTOR message and wait for the reply. If a reply is received within two minutes, set the &ZREVATTENDED variable to ATTENDED (see Figure E-10).

Figure E-10. EVVARSET Macro Parameter Definition

```
SOLVPROD--- Automation Services : EVVARSET Macro Parameter Definition ---------
Command ===>                                                   Function=BROWSE

. Set EventView Variables ---------------------------------------------------.
|                                                                            |
|      Name       Value                                                      |
| ZREV ATTENDED = ATTENDED                                                   |
| ZREV          =                                                            |
| ZREV          =                                                            |
| ZREV          =                                                            |
| ZREV          =                                                            |
| ZREV          =                                                            |
|                                                                            |
'----------------------------------------------------------------------------'
```

See Also

The section, *EVVARGET*, on page E-19 and the section, *Using EventView Variables*, on page 16-12.

EXECNCL

Purpose

Executes an NCL procedure in line with the process that contains the macro. That is, the process waits for the NCL procedure to end before continuing on to the next step.

Parameters

The parameters for the EXECNCL macro are set in the following fields:

NCL Name
Is a mandatory field that specifies the name of the NCL procedure executed by the macro.

Parameters
Specifies any parameters required by the NCL procedure specified in the NCL Name field.

Example

In the following example, the EXECNCL macro executes the $RMSMTS procedure. The EXECNCL procedure contains a number of variables, which are defined in the Parameters field of this panel.

Figure E-11. EXECNCL Macro Parameter Definition

```
SOLVPROD--- Automation Services : EXECNCL Macro Parameter Definition ----------
Command ===>                                                   Function=BROWSE

.- NCL Procedure Details -----------------------------------------------------.
|                                                                              |
|  NCL Name .... $RMSMTS                                                       |
|                                                                              |
|  Parameters .. SERVICE=GET SYSNAME=&ZRMSYSNAME                               |
|                VERSION=&ZRMVERSION CLASS=17                                  |
```

EXTRACT

Purpose

Extracts a segment from a text string.

Parameters

The following parameters can be set for the EXTRACT macro:

Where to Extract From—Input

Specifies the string from which you want to extract the text segment. This string can contain text and variables.

Variable to hold Extracted String—Output

Specifies the name of the variable in which you want to put the extracted text string.

How to do the Extraction

The following fields determine the text you want to extract. You must specify at least one of the Starting Position and Find Text fields. If you complete both fields, the macro looks first at the value of the Find Text field and then adds the value of the Starting Position field. The result locates the first character of the extracted segment.

Starting Position

Is used in conjunction with the Length to Extract field, the Input field, and the Find Text field (if it is completed) to identify the text string you want to extract.

The Starting Position field accepts a number (*n*) from 1 through 256. The value specifies the first character of the text segment to be extracted.

Length to Extract

Specifies the number of characters you want to extract, effectively indicating the end point of the extracted text string.

The Length to Extract field accepts a number from 1 through 256. The value specifies the number of characters that will be in your extracted string. If a value is not specified in this field, the full string or substring is extracted.

Find Text

If completed, must contain a section of the string specified in the Input field.

The Find Text field accepts a text string from 1 through 53 characters in length. The first character in this text string, plus the value in the Starting Position field, determines the first character of the extracted text. The last character in the extracted text is determined by the value in the Length to Extract field.

Examples

In the following example, the EXTRACT macro is used to extract the text string DEF.

Because no value is specified in the Find Text field, the reference point for the starting position is the first character of the value of the Input field. In this example, the first character of the extract is the 5th character of the Input text string (D), and because the length to extract is 3, the resulting extract is DEF.

Figure E-12. EXTRACT Macro Parameter Definition (Example A)

```
SOLVPROD--- Automation Services : EXTRACT Macro Parameter Definition ----------
Command ===>                                                   Function=BROWSE

.- Where to Extract From -------------------------------------------------------.
| Input .... ABC DEF GHI JKL MNO                                                |
|           (Text String or Variable(s))                                        |
'-------------------------------------------------------------------------------'

.- Variable to hold Extracted String -------------------------------------------.
| Output ... FRED         (Variable Name)                                       |
|                                                                               |
'-------------------------------------------------------------------------------'

.- How to do the Extraction ----------------------------------------------------.
| Starting Position ... 5   (Extraction to begin from this position. (1-256)) |
|                                                                               |
| Length to Extract ... 3   (Extract data for this length. (1-256))           |
|                                                                               |
| Find Text ...........                                                         |
|       (Extraction to begin from the start of this text + Starting Position) |
'-------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

In the following example, the EXTRACT macro is used to extract the text string GHI J.

Because no value has been given to the Starting Position field, the reference point for the starting position is the value of the Find Text field. In this example, the first character of the extract is the first character of the Find Text string (G), and because the length to extract is 5, the resulting extract is GHI J.

Figure E-13. EXTRACT Macro Parameter Definition (Example B)

```
SOLVPROD--- Automation Services : EXTRACT Macro Parameter Definition ----------
Command ===>                                                     Function=BROWSE

.- Where to Extract From -------------------------------------------------------.
| Input .... ABC DEF GHI JKL MNO                                                |
|           (Text String or Variable(s))                                        |
'-------------------------------------------------------------------------------'

.- Variable to hold Extracted String -------------------------------------------.
| Output ... FRED         (Variable Name)                                       |
|                                                                               |
'-------------------------------------------------------------------------------'

.- How to do the Extraction ----------------------------------------------------.
| Starting Position ...     (Extraction to begin from this position. (1-256)) |
|                                                                               |
| Length to Extract ... 5   (Extract data for this length. (1-256))           |
|                                                                               |
| Find Text ........... GH                                                      |
|       (Extraction to begin from the start of this text + Starting Position) |
'-------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

In the following example, the EXTRACT macro is used to extract the text string HI JK.

In this example, a value has been given to the Find Text field, and therefore that value becomes the reference point for the starting position. Because a value of 2 has been given to the Starting Position field, the first character of the extract is the second character of the Find Text string (H), and because the length to extract is 5, the resulting extract is HI JK.

Figure E-14. EXTRACT Macro Parameter Definition (Example C)

```
SOLVPROD--- Automation Services : EXTRACT Macro Parameter Definition ----------
Command ===>                                                   Function=BROWSE

.- Where to Extract From -------------------------------------------------.
| Input .... ABC DEF GHI JKL MNO                                           |
|           (Text String or Variable(s))                                   |
'---------------------------------------------------------------------------'

.- Variable to hold Extracted String -------------------------------------.
| Output ... FRED         (Variable Name)                                  |
|                                                                          |
'---------------------------------------------------------------------------'

.- How to do the Extraction ----------------------------------------------.
| Starting Position ... 2   (Extraction to begin from this position. (1-256)) |
|                                                                          |
| Length to Extract ... 5   (Extract data for this length. (1-256))        |
|                                                                          |
| Find Text ...........GH                                                  |
|       (Extraction to begin from the start of this text + Starting Position) |
'---------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

GENALERT

Purpose

Generates an alert for display on the alert monitor. The parameter definition comprises two panels. You must complete the Reference field on the Alert Attributes panel and the Alert Text field on the Alert Definition panel.

If you need to implement a process that generates variations of an alert, you can pass new GENALERT parameter values to the process to create the variations. The passed values override the values specified in the macro. You do not need to create a process for each variation of the alert.

You can specify what actions to take when the alert is raised. To specify actions, press F10 (Actions).

Parameters

The following parameters can be set for the GENALERT macro:

Reference

Identifies the alert.

To override this value, specify ALRTREF=*alert-reference* as a process parameter.

Severity

Specifies the importance of the alert. The valid values are 1 through 4. Severity 1 alerts are most important and severity 4 alerts are least important.

To override this value, specify ALRTSEV=*alert-severity* as a process parameter.

Origin

Specifies the source of the alert.

To override this value, specify ALRTORIGIN=*alert-origin* as a process parameter.

Alert Description

Specifies the text that describes the alert condition.

To override this value, specify ALRTTEXT=*alert-text* as a process parameter.

Alert Text

Provides a detailed description of the alert condition.

To override this value, specify ALRTDESC*n=description-line-n* (where *n* is 1 through 5, representing the five lines of the description) as process parameters for the lines you want to override.

Alert Recommended Action

Specifies the actions that can be performed to remove the alert condition.

To override this value, specify ALRTACTN*n=action-line-n* (where *n* is 1 through 4, representing the four lines of the recommended action) as process parameters for the lines you want to override.

Example

In the following example, the GENALERT macro generates the following UPSALERT9001 alert to remind the operator to test the uninterruptable power supply (UPS):

```
Test UPS
```

Figure E-15. GENALERT Macro Parameter Definition

```
SOLVPROD--------------- EventView : Alert Attributes --------------------------
Command ===>                                                   Function=BROWSE

. Alert Reference Key -------------------------------------------------------.
|                                                                            |
| Reference ... UPSALERT9001                                                 |
|                                                                            |
'----------------------------------------------------------------------------'
. Alert Attributes ----------------------------------------------------------.
|                                                                            |
| Severity .... 4                                                            |
| Origin ...... ALERTMACRO                                                   |
```

```
SOLVPROD--------------- EventView : Alert Definition --------------------------
Command ===>                                                   Function=BROWSE

. Alert Description ---------------------------------------------------------.
|                                                                            |
| TEST UPS                                                                   |
'----------------------------------------------------------------------------'
. Alert Text ----------------------------------------------------------------.
|                                                                            |
| UPS MAINTENANCE                                                            |
|                                                                            |
|                                                                            |
|                                                                            |
|                                                                            |
'----------------------------------------------------------------------------'
. Alert Recommended Action --------------------------------------------------.
|                                                                            |
| TEST THE UPS WHEN THERE ARE NO CRITICAL SYSTEM ACTIVITIES.                 |
|                                                                            |
|                                                                            |
|                                                                            |
'----------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=Exit
 F7=Backward               F9=Swap     F10=Actions  F11=Panels
```

See Also

The section, *DELALERT*, on page E-16.

GETSTAT

Purpose

Discovers the status of a resource or service.

Parameters

The following parameters can be set for the GETSTAT macro:

System Name
Version

Identify the system image that owns the resource. If you do not specify an image, then during execution, the macro uses the following values:

- If the macro is associated with a resource definition, the macro uses the image that owns the definition.
- If the macro is not associated with a resource definition (for example, in an EventView rule), the macro uses the active local image.

Resource Class

Specifies the class name of the resource or service for which you require the status. This field can be completed manually, by entering the appropriate valid code, or automatically, by selecting a value from the prompted field value list. The list is displayed by entering **?** in the Resource Class field.

Resource Name

Specifies the name of the resource or service for which you want to retrieve the status.

Variable Prefix

Specifies the prefix of the variables that contain the retrieved status information. This prefix enables you to store the status information retrieved at different times in different variables.

The information is retrieved from the &ZRMST prefixed status variables. The retrieved information is stored in a corresponding set of variables, with the ZRMST prefix replaced by the specified prefix, *prefix*. For example, the information retrieved from the &ZRMSTACTSTAT variable is stored in the &*prefix*ACTSTAT variable.

Table D-2 on page D-9 lists the status variables.

The prefix must *not* start with one of the following characters: # or $.

User Keywords

Enables you to store the keyword that is specified in the User Defined Keyword field on the Define Extended Display Attribute panel of a triggered message rule in a resource definition. The keyword is stored in the &*prefix*USRKEYW variable.

Example

In the following example, the GETSTAT macro is used to retrieve the status of the STC class resource SOLVIMS into variables of the form &ZRMST*.

Figure E-16. GETSTAT Macro Parameter Definition

```
SOLVPROD--- Automation Services : GETSTAT MACRO Parameter Definition ----------
Command ===>                                                     Function=BROWSE

.- Getstate Key Definition -----------------------------------------------------.
|                                                                                |
|  System Name .............+                                                   |
|  Version .................+                                                   |
|  Resource Class...........+ STC                                               |
|  Resource Name ........... SOLVIMS                                            |
|                                                                                |
'--------------------------------------------------------------------------------'
.--------------------------------------------------------------------------------.
|                                                                                |
|  Variable Prefix........... ZRMST                                             |
|                                                                                |
|  User Keywords ........... NO            (YES/NO)                             |
|                                                                                |
'--------------------------------------------------------------------------------'



 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

GOTO

Purpose

Specifies the next process step to run.

Parameters

The following parameters can be set for the GOTO macro:

Target Step Label
Specifies the name that identifies the next process step to run.

Loop Control Limit
Specifies the runaway loop control limit. It prevents the process from being stuck in a loop.

Example

In the following example, the GOTO macro is used to specify that the YES process step is to run next.

Figure E-17. GOTO Macro Parameter Definition

```
SOLVPROD---- Automation Services : GOTO Macro Parameter Definition ------------
Command ===>                                                   Function=UPDATE

. Goto Target ---------------------------------------------------------------.
|                                                                            |
|  Target Step Label .... STEP3___                                           |
|                                                                            |
|  Loop Control Limit ... 100                                                |
|                                                                            |
'----------------------------------------------------------------------------'
```

PARSE

Purpose

Segments a text string and puts the segmented contents in variables.

Parameters

The parameters for the PARSE macro are set in the following fields:

String to be Segmented—Input

Specifies the string that you want to segment. This string can contain text and variables.

This is a mandatory field. The string can be from 1 through 64 characters in length.

Variables to be Created—Output

Specifies the name of the variables that are to hold your text segments.

This is a mandatory field that can hold from 1 through 64 characters. The names must be in one of the following formats:

- *varname*—use this format when you want to extract one segment only.
- *varname1*,*varname2*,...,*varnamen*—use this format when you want to extract more than one segment, when the segments are separated by delimiter characters such as a space.
- *varname1*(*length1*),*varname*2(*length2*),...,*varnamen*(*lengthn*)—use this format when you want to extract segments delimited by length.
- *prefix***—use this format when you want to put extracted segments into variables &*prefix*1 through &*prefix*64.

If you want to skip a portion of the input string during parsing, specify *(*n*) where *n* represents the number of units to skip. An asterisk by itself is the same as *(1). Depending on the value in the By Length field, skipping occurs as follows:

- If the value is NO, *n* segments are skipped.
- If the value is YES, *n* characters are skipped.

How to Segment Input String

The following fields determine how to segment the text string.

By Length

Indicates whether you want the segmentation to be by length.

This field accepts a YES or NO response. The default response is NO which indicates that the string is to be segmented first by delimiter and then by length. A YES response indicates that segmentation is to be by length only.

By Delimiter

Specifies the characters that are to be used as delimiters for text segments. You can specify up to 8 alphanumeric or special characters. The default character is a space.

Note

If you want to define a series of two characters as a delimiter, those characters must be enclosed in quotation marks.

Preserve Leading and Trailing Blanks?

Specifies whether you want leading and trailing blank spaces to be retained after segmentation. If this field is not completed, the default response of NO is assumed.

Create Null Variables if Consecutive Delimiters Found?

Specifies whether you want a variable (with a null value) to be created if consecutive delimiters are encountered. If this field is not completed, the default response of YES is assumed.

Remainder String Variable Name

Specifies the name of the variable that is to hold the text segment that remains after the specified parsing has been completed.

This field can contain a name from 1 through 12 characters in length.

Example

In the following example, the PARSE macro is used to segment the ABC 123 456 789 input string. The delimiter is the default (space). The segments are placed in the &FRW* variables.

The results will be:

- The variable &FRW1 contains the segment ABC
- The variable &FRW2 contains the segment 123
- The variable &FRW3 contains the segment 456
- The variable &FRW4 contains the segment 789

Figure E-18. PARSE Macro Parameter Definition

```
SOLVPROD---- Automation Services : PARSE Macro Parameter Definition -----------
Command ===>                                                     Function=BROWSE

.- String to be Segmented -----------------------------------------------------.
| Input .... ABC 123 456 789                                                   |
|            (Text String or Variable(s))                                      |
'------------------------------------------------------------------------------'
.- Variables to be Created ----------------------------------------------------.
| Output ... FRW*                                                              |
|            (name or name1,name2... or name1(len),name2(len).. or prefix*)    |
'------------------------------------------------------------------------------'
.- How to Segment the Input String --------------------------------------------.
|    By Length ......          (YES or NO)                                     |
| OR By Delimiter ...          (Eight separate characters may be specified)    |
|                                                                              |
| Preserve Leading and Trailing Blanks? ..................... NO  (YES or NO) |
|                                                                              |
| Create NULL Variables if consecutive delimiters found? .... YES (YES or NO) |
|                                                                              |
| Remainder String Variable Name (If remainder required) ....                  |
'------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

REPLY

Purpose

Replies to an outstanding write-to-operator with reply (WTOR) message. The reply can be to one of the following:

- The last WTOR message
- The first WTOR message from a particular job
- A specified WTOR message

The REPLY macro enables you to specify message rules to detect any expected response messages.

Parameters

The parameters for the REPLY macro are set in the following fields:

WTOR Identification

The following fields identify the WTOR message to reply to.

Last WTOR?

Enter YES in the Last WTOR? field to indicate that the message in the Reply Text field is in response to the last WTOR message seen by this process (for example, a WTOR message sent by a previous step). If the message is not intended as a reply to the last WTOR, leave the field blank.

Note

If your reply is intended for the last WTOR message, complete the Last WTOR field only. Do *not* complete either the Jobname or WTOR Message Text fields.

Jobname

If the message in the Reply Text field is intended as a reply to a WTOR message from a particular job, enter the name of the job in the Jobname field.

If you also complete the WTOR Message Text field, your reply is to that message received from the named job. If you do not complete the WTOR Message text field, your reply is to the first WTOR message received from the named job.

Note

If you complete this field, you cannot also complete the Last WTOR? field.

WTOR Message Text

Specifies the text of the message to which you want to reply. It need not contain the complete message, but must contain sufficient text to adequately identify the message.

If you also complete the Jobname field, your reply is to the WTOR message from the named job only.

Note

If you complete this field, you cannot also complete the Last WTOR? field.

Reply

Specifies your reply. Your reply message can be from 1 through 65 characters in length.

Response Message Analysis

The following fields specify the rules that wait for the response messages to the reply. You can specify up to five rules. The action codes enable you to refine the message rule definition criteria. More information on the use of action codes is available in the online help.

Message Text

Specifies the initial part of the expected response message. It can contain up to 45 characters.

R/Code

Specifies the value to be returned by the macro when it receives the response message. The return code must be a number from 0 through 9999.

E/Filter

Displays YES or NO indicating whether an extended filter has been defined. This field is completed automatically when a rule is defined. It cannot be modified.

Appendix F, *Message Text Analysis*, provides more information about how to define extended message filters.

Wait Time

Specifies the maximum time, in seconds, that the region waits for a response message. The time is reset each time a response is received. The macro waits until a message that matches a rule in the Message Text fields is received or until the time expires without receiving any more responses.

This field must contain a number from 1 through 9999. The default is 30.

Wait Time Expiry Return Code

Sets the return code if the expected message does not arrive before the expiry of the time specified in the Wait Time field.

This field must contain a number from 1 through 999. The default is 69.

Returned Variables

Variables	Description
&$RMMCMSGTEXT	Contains the full text of the message that satisfies the message rules.
&$RMMCWORD*n*	Contain the words in the message, where *n* indicates the position of the word in the message.
&ZRMMSG*n*	Contain any messages received before the required response message, where *n* is a number indicating the order in which those messages are received.
&$RMMCWTORID	Contains the ID of the last encountered WTOR message if the macro encounters any WTOR messages before it encounters an expected message.

For example, if the response to a command consists of three messages A, B, and C, and message B satisfies the message rules, then:

- &ZRMMSG1 contains message A
- &$RMMCMSGTEXT contains message B
- The macro ends on receipt of message B, and does not see message C

Example

In the following example, the REPLY macro is used to reply to the WTOR message identified by the following text string:

```
$HASP100 SOLVVLF ON STCINRDR
```

Figure E-19. REPLY Macro Parameter Definition

```
SOLVPROD---- Automation Services : REPLY Macro Parameter Definition -----------
Command ===>                                                   Function=UPDATE

.- WTOR Identification -------------------------------------------------------.
| Last WTOR? Or Jobname And/Or WTOR Message Text                               |
|    ___ (Yes)  ________       $HASP100 SOLVVLF  ON STCINRDR________________ |
'------------------------------------------------------------------------------'
.- Reply Text ----------------------------------------------------------------.
|                                                                              |
| Reply.... IEFTMS81 CA-1 WTO INTERCEPT ACTIVATED______________________________ |
'------------------------------------------------------------------------------'
.- Response Message Analysis -------------------------------------------------.
|                                 D=Delete Extended Filter  S=Extended Filter |
|           Message Text                                     R/Code  E/Filter |
|   ___     ______________________________________________     ___             |
|   ___     ______________________________________________     ___             |
|   ___     ______________________________________________     ___             |
|   ___     ______________________________________________     ___             |
|   ___     ______________________________________________     ___             |
|                                                                              |
| Wait Time 30__  Wait Time Expiry Return Code 69_                             |
```

RUNPRCSS

Purpose

Runs another process in line with the current process. The current process waits for the other process to end before continuing to the next step.

Parameters

The parameters for the RUNPRCSS macro are set in the following fields:

Process
Specifies the name of the process to be executed by the macro.

Parameters
Only relevant if the process specified in the Process field contains variables that require you to pass values to them.

If the specified process does not require any passed values, leave this field empty.

Example

In the following example, the RUNPRCSS macro runs the STPRT process. The STPRT process contains the &MODEL and &NAME variables that are defined in the Parameters field of this panel.

Figure E-20. RUNPRCSS Macro Parameter Definition

```
SOLVPROD--- Automation Services : RUNPRCSS Macro Parameter Definition ---------
Command ===>                                                   Function=BROWSE

.- Process Details -------------------------------------------------------.
|                                                                          |
|  Process ..... STPRT                                                     |
|                                                                          |
|  Parameters .. MODEL=HPL36 NAME="This is a name"                         |
|                                                                          |
|                                                                          |
'--------------------------------------------------------------------------'
```

SETRC

Purpose

Sets the process return code.

The return code is the value that is returned by a macro when each process step is completed. This is the trigger for the next step to begin, provided that any specified conditions are met.

Parameters

The SETRC macro has only one parameter: the return code.

The value of this parameter must be a number from 0 through 99.

Example

In the following example, the SETRC macro is used to set the return code to 21.

Figure E-21. SETRC Macro Parameter Definition

```
SOLVPROD---- Automation Services : SETRC Macro Parameter Definition -----------
Command ===>                                                   Function=BROWSE

.- Return Code -------------------------------------------------------------.
|                                                                            |
|  Return Code .... 21  (Enter a number between 0 and 99)                    |
|                                                                            |
'----------------------------------------------------------------------------'
```

SETSTATE

Purpose

Sets the actual state and the extended display attributes of a resource or service.

Parameters

The parameters for the SETSTATE macro are set in the following fields:

System Name
Version

Identify the system image that owns the resource. If you do not specify an image, then during execution, the macro uses the following values:

- If the macro is associated with a resource definition, the macro uses the image that owns the definition.
- If the macro is not associated with a resource definition (for example, in an EventView rule), the macro uses the active local image.

Resource Name

Specifies the resource or service for which you are setting the state. The value of this field can be a variable whose value is passed to this macro by the process itself.

This field is mandatory.

Resource Class

Specifies the class of the item named in the Resource Name field.

Entering a question mark (?) in the Resource Class field displays the Resource Class List panel from which you can select the appropriate class.

State

Specifies the actual state you want to set for the named resource or service.

This field is mandatory and must contain one of the following (entered either directly or by using a variable):

ACTIVE	DEGRADED	FAILED	
INACTIVE	STARTING	STOPPING	UNKNOWN

Description

Specifies the free form text that can be used to identify who has set the state or the reason for setting that state. The value in this field is displayed as part of the detailed status information, which is accessed from a monitor.

Extended Display

The following fields are used to set the attributes of the extended display that overlays the current status display.

Intensity

Specifies the intensity of the display text and must be either HIGH or LOW. If no value is set, the default value of LOW is assumed.

Color

Specifies the color of the display text and must be one of BLUE, GREEN, PINK, RED, TURQUOISE, WHITE, or YELLOW. If no value is set, the default value of GREEN is assumed.

Highlight

Specifies the highlight used for the display text and must be one of BLINK, REVERSE, or NONE. If no value is set, the default value of NONE is assumed.

Icon Flag

Specifies whether the extended display color of a resource that is in the worst logical state should be used on an icon.

If you do not want the icon to use extended display color, specify **NO** (the default) in this field. For example, if the resource does not require attention, transferring the color to the icon can cause an undesirable change in the icon color.

If you want the icon to use extended display color, specify **YES** in this field. This setting also forces the state ranking of the resource to equal that of the FAILED logical state. The icon can thus be forced to display this resource even though it is not in the worst logical state. For example, if you rate the condition of a tape mount request very important, specify **YES** in the Use on Graphic Monitor? field.

Text

Specifies the free form extended status text that overlays the current display.

If you do not want to change the status display, leave the field blank.

If you want to reset the status line to normal display, type ##RESET## in this field.

Example

In the following example, the SETSTATE macro is used to set the actual state and the extended display according to the parameters passed to the process that contains the macro.

Figure E-22. SETSTATE Macro Parameter Definition

```
SOLVPROD--- Automation Services : SETSTATE Macro Parameter Definition ---------
Command ===>                                                   Function=BROWSE

.- Resource Identification -----------------------------------------------------.
| System Name ......+                                                          |
| Version ..........+                                                          |
| Resource Class ...+ &CLS                                                     |
| Resource Name ..... &NAME                                                    |
'------------------------------------------------------------------------------'
.- Actual State ----------------------------------------------------------------.
| State ............ &V1      (Active, Inactive, Failed, Degraded,             |
|                              Starting, Stopping or Unknown)                  |
| Description ...... &REASON                                                   |
'------------------------------------------------------------------------------'
.- Extended Display ------------------------------------------------------------.
| Intensity ... LOW        (High or Low)                                       |
| Color ....... GREEN      (Blue, Green, Pink, Red, Turquoise, White, Yellow) |
| Highlight ... NONE       (Blink, Reverse or None)                            |
| Icon Flag ...            (Use Attributes to change Icon color - Yes or No)  |
|                                                                              |
| Text ........ &EXTDISP                                                       |
'------------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

SETVARS

Purpose

Creates variables for a process.

The SETVARS macro stores the values of transient variables. For example, the values of the &$RMMCWORD*n* variables change for each executed COMMAND, REPLY, and SYSCMD macro. Using the SETVARS macro, you can store the original values before they change.

Parameters

The SETVARS macro has only one parameter: the string of variables to be created.

The Variable String field defines the variables (and their values) required by the process. These variables are passed to all macros within the process.

Example

In the following example, the SETVARS macro is used to set the value of the &V1 and &V2 variables.

Figure E-23. SETVARS Macro Parameter Definition

```
SOLVPROD--- Automation Services : SETVARS Macro Parameter Definition ----------
Command ===>                                                   Function=BROWSE

.- Variable Details -------------------------------------------------------.
|                                                                           |
|  Enter the variables to be created below. (e.g. VAR1=TEST VAR2=&$RMMCWORD1) |
|                                                                           |
|  Variable String... V1=ACTUAL V2=INACTIVE                                 |
|                                                                           |
|                                                                           |
|                                                                           |
'---------------------------------------------------------------------------'
```

SMFWRITE

Purpose

Writes a user-defined SMF record (SMS record on VOS3 systems).

Parameters

The parameters for the SMFWRITE macro are set in the following fields:

Test Mode
Can be used to test the macro.

If the test mode is OFF, the macro writes the specified record to the SMF (or SMS) dataset. If the test mode is ON, the macro sends the specified record to the activity log.

Subtype
Specifies the record subtype. The 9*xxx* subtypes are reserved for these user-defined records.

Enter a hexadecimal value. The subtype is constructed by prefixing the specified value with 9.

SMF Data
Specifies the data in each field of the record. This field may contain constant or variable data (data contained in variables).

Length
Specifies the length of a data field in bytes.

Example

In the following example, the SMFWRITE macro is used to write a record that indicates when a service starts.

Figure E-24. SMFWRITE Macro Parameter Definition

```
--------- Automation Services : SMFWRITE Macro Parameter Definition -----------
Command ===>                                                   Function=BROWSE

.- SMF Parameters -------------------------------------------------------------.
|                                                                               |
| Test Mode ... OFF                                                             |
| Subtype ..... FFF                                                             |
|                                                                               |
|   SMF Data                                                            Length  |
|   START                                                                 6     |
|   &SERVICE                                                              12    |
```

SNMPTRAP

Purpose

Generates an SNMP trap. You can use an SNMP trap to inform a remote system of a resource or service state change.

Parameters

The parameters for the SNMPTRAP macro are set in the following fields:

Text
Specifies the text to send with the trap. The format of the text should conform to the specified enterprise ID and specific trap number.

Destinations Dataset
Specifies the dataset name (DSN) in which the address(es) of remote system(s) are located.

Destination Address(es)
Specifies the destination address(es) of the remote system(s). You can enter an address, or multiple addresses by using a comma or a space as the delimiting character.

Enterprise ID
Together with the specific trap number, specifies the format of the trap. The format identified by this field is determined by your open platform administration.

Specific Trap Number
Together with the enterprise ID, specify the format of the trap. The format identified by this field is determined by your open platform administration.

Community Name
Specify the community for which the trap is destined. This field is *case sensitive* and can be left with its default of `public`.

Example

In the following example, the SNMPTRAP macro sends a trap to the remote system at the specified address when the CICS task is lost. The macro is specified in a process definition that is used by the resource definition for the task (CICSPROD) and processed as an exit upon receipt of a message which indicated that the CICS region has failed.

Figure E-25. SNMPTRAP Macro Parameter Definition

```
---------- Automation Services : SNMPTRAP Macro Parameter Definition ----------
Command ===>                                                   Function=UPDATE

. SNMP Trap Details -------------------------------------------------------.
|                                                                          |
| Text ..... SOLVE1_____________________________________________________ |
|            SOLVE2_____________________________________________________ |
|            SOLVE3_____________________________________________________ |
|            SOLVPROD-0001______________________________________________ |
|            CICS(CICSPROD)_____________________________________________ |
|            SOLVPROD-0001: CICSPROD HAS FAILED_________________________ |
|                                                                          |
| Destinations Dataset ...... ____________________________________________ |
|            or                                                            |
| Destination Address(es) ... 128.1.170.23______________________________ |
|                                                                          |
| Enterprise Id ............. 1.3.6.1.4.1.11.2.17.1_____________________ |
|                                                                          |
| Specific Trap Number ...... 59047936__________________________________ |
|                                                                          |
| Community Name ............ PUBLIC_____________________________________ |
'--------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=OK
                           F9=Swap                                F12=Cancel
```

STARTNCL

Purpose

Starts an NCL procedure that runs independently of the process.

Parameters

The parameters for the STARTNCL macro are set in the following fields:

NCL Name
Specifies the name of the NCL procedure you want to start.

Region
Identifies the region in which the specified NCL procedure is to start. The field must be blank or contain one of the following background environments:

AOMP BLOG BMON BSYS CNMP LOGP PPOP

If this field is blank, the current region is assumed.

Parameters
Specifies the parameters (and their values) required by the NCL procedure specified in the NCL name field.

Example

In the following example, the STARTNCL macro starts the CALC2 NCL procedure which contains the &MULT variable. The procedure is started in the current region.

Figure E-26. STARTNCL Macro Parameter Definition

```
SOLVPROD--- Automation Services : STARTNCL Macro Parameter Definition ---------
Command ===>                                                   Function=BROWSE

.- NCL Procedure Details -------------------------------------------------------.
|                                                                                |
|  NCL Name .... CALC2                                                           |
|                                                                                |
|  Region .....+           (AOMP, BLOG, BMON, BSYS, CNMP, LOGP or PPOP)          |
|                                                                                |
|  Parameters .. MULT=365                                                        |
|                                                                                |
|                                                                                |
|                                                                                |
|                                                                                |
'--------------------------------------------------------------------------------'
```

STOP

Purpose

Stops a process after setting the return code.

Parameters

The STOP macro has only one parameter: return code.

The Return Code field enables you to set the process return code. This is a mandatory field and accepts a numeric value from 0 through 99.

Example

In the following example, the return code for the STOP macro is set to 0.

Figure E-27. STOP Macro Parameter Definition

```
SOLVPROD---- Automation Services : STOP Macro Parameter Definition ------------
Command ===>                                                   Function=BROWSE

.- Return Code -------------------------------------------------------------.
|                                                                            |
|  Return Code .... 0   (Enter a number between 0 and 99)                    |
|                                                                            |
'----------------------------------------------------------------------------'
```

SUBJOB

Purpose

Submits a JCL batch job.

Parameters

The parameters for the SUBJOB macro are set in the following fields:

JCL Details

The following fields are used to define the job submitted and the return code details.

Dataset Name

Is a mandatory field that specifies the dataset member that contains the job steps.

Wait Time

Specifies the maximum time, in seconds, that the region waits for a response message. The time is reset each time a response is received. The macro waits until a message that matches a rule in the Message Text fields is received or until the time expires without receiving any more responses.

This field must contain a number from 1 through 9999. The default is 30.

Wait Time Expiry Return Code

Specifies the value of the macro return code if the expected message does not arrive before the expiry of the time specified in the Wait Time field.

This field must contain a number from 1 through 999. The default is 69.

Response Message Analysis

The following fields specify the rules that determine the required response messages. You can specify up to five rules. The action codes enable you to refine the message definition criteria. More information on the use of action codes is available from the online help.

Message Text

Specifies the initial part of the required response message. It can contain up to 45 characters.

Return Code

Specifies the value set by the macro when it receives the response message. The return code must be a number from 0 through 999.

Extended Filter?

Indicates whether an extended filter has been defined. This field is empty until you complete the Message Text and Return Code fields. If you want to define an extended filter, enter **S** to access the Extended Message Filtering panel.

After this panel is completed, the Extended Filter field is updated automatically.

Appendix F, *Message Text Analysis*, provides more information about how to define extended message filters.

Returned Variables

Variables	Description
&$RMMCMSGTEXT	Contains the full text of the message that satisfies the message rules.
&$RMMCWORD*n*	Contain the words in the message, where *n* indicates the position of the word in the message.
&ZRMMSG*n*	Contain any messages received before the required response message, where *n* is a number indicating the order in which those messages are received.
&$RMMCWTORID	Contains the ID of the last encountered WTOR message if the macro encounters any WTOR messages before it encounters an expected message.

For example, if the response to a command consists of three messages A, B, and C, and message B satisfies the message rules, then:

- &ZRMMSG1 contains message A
- &$RMMCMSGTEXT contains message B
- The macro ends on receipt of message B, and does not see message C

Example

In the following example, the name of the dataset member that contains the job steps is SOLV0.RAM.TEMPEXEC(JPTAPE).

Figure E-28. SUBJOB Macro Parameter Definition

```
SOLVPROD -- Automation Services : SUBJOB Macro Parameter Definition -----------
Command ===>                                                   Function=BROWSE

.- JCL Details -------------------------------------------------------------.
|                                                                            |
| Dataset Name .... SOLV0.RAM.TEMPEXEC(JPTAPE)____________________________   |
|                                                                            |
| Wait Time ... 0     Wait Time Expiry Return Code ... 69                    |
'----------------------------------------------------------------------------'
.- Response Message Analysis -----------------------------------------------.
|                                                          S=Extended Filter |
|           Message Text                                     Return Extended |
|                                                            Code   Filter?  |
|   ___                                                                      |
|   ___                                                                      |
|   ___                                                                      |
|   ___                                                                      |
|   ___                                                                      |
|                                                                            |
'----------------------------------------------------------------------------'


 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

SUBPRCSS

Purpose

Submits another process to a region. The process runs separately from the current process—that is, *not* in line with the current process.

Parameters

The parameters for the SUBPRCSS macro are set in the following fields:

Process
Specifies the name of the process you want to start.

Region
Identifies the owner of the processing region in which the process is to start. In this case, the owner is a background user and must be one of:

AOMP BLOG BMON BSYS CNMP LOGP PPOP

This is a mandatory field.

Log
Indicates the way in which you want the process activity logged to the activity log. The value of this field must be one of the following:

- NO (no log)
- FULL (full log)
- SUMM (summary log)
- BOTH (both full and summary logs)

If this field is not completed, the default value of NO is assumed.

Parameters
Specifies the parameters (and their values) required by the specified process.

Example

In the following example, the process STAMP has been set to run in the BLOG environment. The activities of STAMP are fully logged. STAMP contains the &NAME variable which is defined as WTOH.

Figure E-29. SUBPRCSS Macro Parameter Definition

```
SOLVPROD--- Automation Services : SUBPRCSS Macro Parameter Definition ---------
Command ===>                                                     Function=BROWSE

.- Process Details -------------------------------------------------------.
|                                                                          |
|  Process ..... STAMP                                                     |
|                                                                          |
|  Region .....+ BLOG      (AOMP, BLOG, BMON, BSYS, CNMP, LOGP or PPOP)    |
|                                                                          |
|  Log ........+ FULL      (NO, FULL, SUMM, BOTH)                          |
|                                                                          |
|  Parameters .. NAME=WTOH                                                 |
|                                                                          |
|                                                                          |
'--------------------------------------------------------------------------'
```

SVAPI

Purpose

Executes the $RMCALL API. The macro inherits the $RMCALL return code. See the *Automation Services Administrator Guide* for a description of the call.

Parameters

The parameters for the SVAPI macro are set in the following fields:

Opt

Refers to the requested option and is automatically set to SERVICE.

Service

Refers to the requested SERVICE and is automatically set to ACTION.

Action

Mandatory field that identifies the type of processing required. The value of this field must be one of the following:

- COMMAND issues an Automation Services command.
- DBGET retrieves the definition of a resource or service specified by the SysName, Version, Class, and Name fields. The information is stored in the &ZRMDB prefixed variables.
- STGET retrieves the current status of the resource or service specified by the SysName, Version, Class and Name fields. The information is stored in the &ZRMST prefixed variables.

Command

Identifies the required command if the value of the Action field is COMMAND.

SysName

Identifies the system image in which to perform the action. If this field is not completed, the current system image is assumed by default.

Version

Identifies the version of the system image in which to perform the action. If this field is not completed, the current system image is assumed by default.

Class

Identifies the resource or service class. See Appendix H, *Knowledge Base Definition Classes*, for information about the classes.

Name

Identifies the resource or service.

Parms

Specifies the parameters required by the command specified in the Command field.

Example

In the following example, the macro is used to set the actual state of the local CICSA resource to ACTIVE.

Figure E-30. SVAPI Macro Parameter Definition

```
SOLVPROD---- Automation Services : SVAPI Macro Parameter Definition -----------
Command ===>                                                     Function=UPDATE

.- API Parameters -------------------------------------------------------------.
|                                                                               |
|  Opt ........ SERVICE              Service .... ACTION                        |
|                                                                               |
|  Action ..... COMMAND              Command .... ASA                           |
|                                                                               |
|  SysName .... ________             Version .... ____                          |
|                                                                               |
|  Class ...... 02                   Name ....... CICSA                         |
|                                                                               |
|  Parms ...... ________________________________________________________________ |
|                                                                               |
'-------------------------------------------------------------------------------'




 F1=Help      F2=Split     F3=OK
                           F9=Swap                                F12=Cancel
```

SVCMD

Purpose

Issues a registered Automation Services command. The macro inherits the $RMCALL return code.

Parameters

The parameters for the SVCMD macro are set in the following fields:

Command Name
Is a mandatory field that specifies the name of the registered command issued by the SVCMD macro.

To display a full list of registered commands, enter **?**.

Parameters
Specifies the parameters passed to the specified command.

Example

The following example issues the T (Terminate) command for the local resource named in &NAME.

Figure E-31. SVCMD Macro Parameter Definition

```
SOLVPROD---- Automation Services : SVCMD Macro Parameter Definition -----------
Command ===>                                                   Function=BROWSE

.- Command Details ----------------------------------------------------------.
|                                                                            |
|  Command Name  T                                                           |
|                                                                            |
|  Parameters .. SYSNAME=&ZRMSYSNAME VERSION=&ZRMVERSION CLASS=17            |
|                NAME=&NAME                                                  |
```

SYSCMD

Purpose

Issues a system command and interprets the results.

The SYSCMD macro enables you to specify message rules to detect the required response message.

Parameters

The parameters for the SYSCMD macro are set in the following fields:

System Command

The following fields are used to define the command sent, the job from which a response is to be acknowledged, and the return code details.

Command

Is a mandatory field that specifies the system command issued by the SYSCMD macro.

The command follows the syntax of the SYSCMD command. See the online help or the *Command Reference* for information about the SYSCMD command.

Jobname

Enables you to limit the response messages to those from the specified job.

Wait Time

Specifies the maximum time, in seconds, that the region waits for a response message. The time is reset each time a response is received. The macro waits until a message that matches a rule in the Message Text fields is received or until the time expires without receiving any more responses.

This field must contain a number from 1 through 9999. The default is 30.

Wait Time Expiry Return Code

Specifies the value of the macro return code if the expected message does not arrive before the expiry of the time specified in the Wait Time field.

This field must contain a number from 1 through 999. The default is 69.

Response Message Analysis

The following fields specify the rules that determine the required response messages. You can specify up to five rules. The action codes enable you to refine the message definition criteria. More information on the use of action codes is available from the online help.

Message Text

Specifies the initial part of the required response message. It can contain up to 45 characters.

Return Code

Specifies the value set by the macro when it receives the response message. The return code must be a number from 0 through 999.

Extended Filter?

Indicates whether an extended filter has been defined. This field is empty until you complete the Message Text and Return Code fields. If you want to define an extended filter, enter **S** to access the Extended Message Filtering panel.

After this panel is completed, the Extended Filter field is updated automatically.

Appendix F, *Message Text Analysis*, provides more information about how to define extended message filters.

Returned Variables

Variables	Description
&$RMMCMSGTEXT	Contains the full text of the message that satisfies the message rules.
&$RMMCWORD*n*	Contain the words in the message, where *n* indicates the position of the word in the message.
&ZRMMSG*n*	Contain any messages received before the required response message, where *n* is a number indicating the order in which those messages are received.
&$RMMCWTORID	Contains the ID of the last encountered WTOR message if the macro encounters any WTOR messages before it encounters an expected message.

For example, if the response to a command consists of three messages A, B, and C, and message B satisfies the message rules, then:

- &ZRMMSG1 contains message A
- &$RMMCMSGTEXT contains message B
- The macro ends on receipt of message B, and does not see message C

Example

In the following example, the SYSCMD macro issues the S DENMIMS1 command and awaits a specified response.

No extended filter has been defined in this example.

Figure E-32. SYSCMD Macro Parameter Definition

```
SOLVPROD--- Automation Services : SYSCMD Macro Parameter Definition -----------
Command ===>                                                   Function=BROWSE

.- System Command -------------------------------------------------------------.
|                                                                               |
| Command ..... S DENMIMS1                                                      |
|                                                                               |
| Jobname .....                                                                 |
|                                                                               |
| Wait Time ... 5     Wait Time Expiry Return Code ... 0                        |
'-------------------------------------------------------------------------------'
.- Response Message Analysis --------------------------------------------------.
|                                                          S=Extended Filter |
|           Message Text                                     Return Extended |
|                                                            Code   Filter?  |
|   ___   IEF403I DENMIMS1 DENMIMS1                             0      NO    |
|   ___   IEF450I DENMIMS1 DENMIMS1                             6      NO    |
|   ___                                                                       |
|   ___                                                                       |
|   ___                                                                       |
|                                                                             |
'-------------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

TRANS

Purpose

Changes all occurrences of a set of characters in a text string into another set of characters.

Parameters

The parameters for the TRANS macro are set in the following fields:

How to Translate

The following fields specify the character translations to perform:

From

Specifies the list of characters you want to translate (replace). You must specify at least one character in this list.

To

Specifies the list of corresponding replacement characters.

String to be Translated—Input

Specifies the string that contains the characters you want translated.

The field is mandatory, and the entry may contain one or more variables. Translation occurs after the variables are substituted by their values.

Variable to hold Translated String—Output

Specifies the name of the variable to hold the translated string.

This is a mandatory field.

Example

In the following example, the TRANS macro examines the Input string:

```
AAA BBB CCC DDD EEE FFF GGG HHH
```

All occurrences of the specified characters are translated so that the result is:

```
111 222 333 444 555 FFF GGG HHH
```

This string is held in the variable &FRED1.

Figure E-33. TRANS Macro Parameter Definition

```
SOLVPROD---- Automation Services : TRANS Macro Parameter Definition -----------
Command ===>                                                     Function=BROWSE

.- What to Translate -------------------------------------------------------.
|                                                                            |
| From ..... A    To ....... 1   (All occurrences of the From character(s)   |
|      ..... B       ....... 2   in the Input string will be translated to   |
|      ..... C       ....... 3   the corresponding To character(s), and the  |
|      ..... D       ....... 4   result will be stored in the Output string) |
|      ..... E       ....... 5                                               |
'----------------------------------------------------------------------------'

.- String to be Translated -------------------------------------------------.
|                                                                            |
| Input .... AAA BBB CCC DDD EEE FFF GGG HHH                                 |
|           (Text String or Variable(s))                                     |
'----------------------------------------------------------------------------'

.- Variable to hold Translated String --------------------------------------.
|                                                                            |
| Output ... FRED1        (Variable Name)                                    |
'----------------------------------------------------------------------------'
 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

WAIT

Purpose

Instructs a process to wait for a specified period before executing the next step in the process.

Parameters

The WAIT macro has only one parameter: the wait time.

This field is mandatory and enables you to specify the length of time (in seconds) that the process is to wait before proceeding.

The value must be a number from 1 through 9999.

Example

In the following example, the WAIT macro is used to instruct the process to wait 20 seconds before proceeding.

Figure E-34. WAIT Macro Parameter Definition

```
SOLVPROD---- Automation Services : WAIT Macro Parameter Definition ------------
Command ===>        :                                          Function=BROWSE

.- Wait Time -------------------------------------------------------------.
|                                                                          |
|  Wait Time .... 20   (Enter a number of seconds between 1 and 9999)      |
|                                                                          |
'--------------------------------------------------------------------------'
```

WAITEVNT

Purpose

Waits for an N00102 event message.

Parameters

The parameters for the WAITEVNT macro are set in the following fields:

Note

You can use the asterisk (*) as wild cards in the Name, Object, Resource, and Reference fields. A leading or embedded * represents a single character; a trailing * represents zero or more characters.

Name

Enables you to use the name of the event as a criterion to select the event to wait for.

Type

Enables you to use the type of event as a criterion to select the event to wait for. Valid values are as follows:

ALL	Selects events irrespective of their contents.
APPLICATION	Selects events that contain user-defined information.
ACCESS	Selects events that contain security alarm information.
CONFIGURATION	Selects events that contain object definition and relationship information.
PROCEDURAL	Selects events that contain scheduling and process control information.
SERVICEABILITY	Selects events that contain availability, degradation, error, fault, and recovery information.
UTILISATION	Select events that contain accounting, performance, response time, and statistics information.

Scope

Enables you to use the scope of delivery of the event as a criterion to select the event to wait for. Valid values are as follows:

REGION	Selects events that are delivered to processes within the region of the event issuer only.
SYSTEM	Selects events that are delivered to processes within the entire region.

Object

Enables you to use the object class as a criterion to select the event to wait for.

Resource

Enables you to use the name of the resource or resource pair (separated by a comma) for which the event is issued as a criterion to select the event to wait for.

Reference

Enables you to use the reference code for the event as a criterion to select the event to wait for.

Wait Time

Specifies how long the macro can wait for a event that satisfies the specified criteria.

Return Codes

Return Code	Description
0	Event message received
4	Event message not received within the specified time

Returned Variables

Variables	Description
&$RMMCEVENT1 through &$RMMCEVENT12	Contain the N00102 event message. The number of variables that contain value depends on the length of the message. Each variable can hold up to 256 bytes.
&$RMMCEVNAME	Contains the name of the event.
&$RMMCEVTYPE	Contains the type of the information in the event.
&$RMMCEVSCOPE	Contains the scope of delivery of the event.
&$RMMCEVOBJ	Contains the object class for the resource or resource pair for which the event is issued.
&$RMMCEVRES	Contains the name of the resource or resource pair for which the event is issued.
&$RMMCEVREF	Contains the reference code for the event.
&$RMMCEVDATA	Contains the part of data that is in the &$RMMCEVENT1 variable. If the data overflows into the other &$RMMCEVENT*n* variables, you can obtain all the data by concatenating the value in those variables to the value in this variable.

Example

In the following example, the WAITEVNT macro waits for an event with a name that starts with OPERATIONS, and of the specified type and scope. The wait time period is 100 seconds.

Figure E-35. WAITEVNT Macro Parameter Definition

```
SOLVPROD--- Automation Services : WAITEVNT Macro Parameter Definition ---------
Command ===>                                                   Function=BROWSE

.- Event Parameters -------------------------------------------------------.
|                                                                           |
| Name......... OPERATIONS*                                                 |
|                                                                           |
| Type........+ SERVICEABILITY                                              |
| Scope.......+ SYSTEM                                                      |
|                                                                           |
| Object.......                                                             |
| (Blanks will be translated to underscores)                                |
|                                                                           |
| Resource.....                                                             |
| Reference....                                                             |
|                                                                           |
| Wait Time ... 100 (0 - 9999, Blank = Wait Forever)                        |
```

See Also

The section, *EVENT,* on page E-17.

WAITSTAT

Purpose

Waits for a specific resource or service state change and retrieves the status.

Parameters

The parameters for the WAITSTAT macro are set in the following fields:

System Name
Version

Identify the system image that owns the resource. If you do not specify an image, then during execution, the macro uses the following values:

- If the macro is associated with a resource definition, the macro uses the image that owns the definition.
- If the macro is not associated with a resource definition (for example, in an EventView rule), the macro uses the active local image.

Resource Class

Specifies the class name of the resource or service for which you want to detect the state change. This field can be completed manually, by entering the appropriate valid value, or automatically by selecting from the prompted field value list. To display the list, type **?** in the Resource Class field. See the section, *GETSTAT*, on page E-30 in this appendix for the contents of the list.

Resource Name

Specifies the name of the resource or service for which you want to detect the state change.

State Type

Specifies whether the specified state change applies to the actual state, desired state, or the logical state. This field can be completed manually, by entering the appropriate valid type of state, or automatically by selecting from the prompted field value list. To display the list, type **?** in the State Type field.

Transition

The following fields specify the state change to wait for:

From State

Specifies the required state before the change. You can specify ANY to detect state changes from any state of the specified type. This field can be completed manually by entering the appropriate valid state, or automatically by selecting from the prompted field value list. To display the list, type **?** in the From State field.

If the current state does not satisfy this value, the macro cannot be processed and the return code is set to 8.

Note

The values listed in the prompted field value list will vary depending on the value of the State Type field.

To State

Specifies the required state after the change. This field can be completed manually by entering the appropriate valid state, or automatically by selecting from the prompted field value list. To display the list, type **?** in the To State field.

Note

The values listed in the prompted field value list will vary depending on the value of the State Type field.

If the value of the From State field is ANY and the current state satisfies this value, the state change condition is deemed to be satisfied and the status is retrieved.

Timeout Period

Specifies the maximum time in seconds that can elapse while waiting for the state change. After the specified time has expired, the return code is set to the value specified in the Return Code field.

Return Code

Specifies the return code to set if timeout occurs. The return code must be a number from 0 through 99. *Do not specify 8 as the return code here*.

Variable Prefix

Specifies the prefix of the variables that contain the retrieved status information. The prefix enables you to store the status information retrieved at different times in different variables.

The information is retrieved from the &ZRMST prefixed status variables when the specified state change is detected. The retrieved information is stored in a corresponding set of variables, with the ZRMST prefix replaced by the specified prefix, *prefix*. For example, the information retrieved from the &ZRMSTACTSTAT variable is stored in the &*prefix*ACTSTAT variable.

Table D-2 on page D-6 describes the status variables.

The prefix must *not* start with one of the following characters: # or $.

User Keywords

Enables you to store the keyword that is specified in the User Defined Keyword field on the Define Extended Display Attribute panel of a triggered message rule in a resource definition. The keyword is stored in the &*prefix*USRKEYW variable.

Example

In the following example, you want to detect a change in the actual state of the resource SOLVIMS from INACTIVE to ACTIVE. If the change is not detected within 30 seconds, the process sets the return code to 16. A variable prefix of ZRMST has been specified.

Figure E-36. WAITSTAT Macro Parameter Definition

```
SOLVPROD--- Automation Services : WAITSTAT MACRO Parameter Definition ---------
Command ===>                                                   Function=BROWSE

.- State Change Event Receipt Profile ----------------------------------------.
|                                                                              |
|  System Name .............+                                                 |
|  Version .................+                                                 |
|  Resource Class ..........+ STC                                             |
|  Resource Name ........... SOLVIMS                                          |
|                                                                              |
+------------------------------------------------------------------------------+
|                                                                              |
|  State Type ..............+ ACTUAL                                          |
|  Transition From State ...+ INACTIVE                                         |
|             To State .....+ ACTIVE                                           |
|  Timeout period ........... 30            (Seconds - 1-9999)               |
|  Return code .............. 16            (0-99)                           |
|  Variable Prefix .......... ZRMST                                            |
|  User Keywords ............ NO            (YES/NO)                         |
|                                                                              |
'------------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

WTO

Purpose

Sends a WTO message to local system consoles.

Parameters

The parameters for the WTO macro are set in the following fields:

WTO Text
Specifies the WTO message to send.

Descriptor Codes
Specifies the descriptor codes assigned to the message. The value can be a single code, or a list of codes and code ranges (for example, 1,8-12).

Routing Codes
Specifies the routing codes assigned to the message. The value can be a single code, or a list of codes and code ranges (for example, 1,8-12).

Example

In the following example, you send a message to consoles profiled for routing codes 1, 2, and 11 to indicate that the region is now managing systems network operations.

Figure E-37. WTO Macro Parameter Definition

```
SOLVPROD------ Automation Services : WTO Macro Parameter Definition -----------
Command ===                                                   Function=BROWSE

.- WTO Details -------------------------------------------------------------.
|                                                                            |
| WTO Text... THE REGION IS NOW MANAGING SYSTEM OPERATIONS.                  |
|                                                                            |
|                                                                            |
| Descriptor Codes...                                                        |
|                                                                            |
| Routing Codes...... 1,2,11                                                 |
`____________________________________________________________________________'
```

WTOR

Purpose

Sends a WTOR message to all local system consoles and waits for the reply.

Parameters

The parameters for the WTOR macro are set in the following fields:

WTOR Text To Be Issued—WTOR Text
Specifies the WTOR message to send.

Response Text Analysis
The following fields specify how to handle replies:

Reply Match Text
Specifies the replies to look for. You can specify up to five reply text versions (rules), each setting a specific value for the macro return code. Specify enough text to positively identify each version. You can use the asterisk (*) as the wildcard character in the reply text.

R/Code
Specifies the value to set for the return code if a reply satisfying the rule in the Reply Match Text field is received. Use different values to differentiate between the different replies. The macro return code thus indicates the type of reply received.

Store Actual Reply Text in Variable Name
Specifies the name of the variable in which to put the received reply.

Wait Time
Specifies the number of seconds to wait for a reply that satisfies one of the rules in the Reply Match Text field. If the specified time expires, the WTOR message is cancelled and the return code is set to the value specified in the Wait Time Expiry Return Code field.

Wait Time Expiry Return Code
Specifies the value to set for the return code if no valid reply is received within the specified time.

Example

In the following example, you want to find out whether one of five specified users has logged on to a console.

Figure E-38. WTOR Issue Macro Parameter Definition

```
SOLVPROD--- Automation Services : WTOR Issue MACRO Parameter Definition -------
Command ===                                                   Function=BROWSE

.- WTOR Text To Be Issued -------------------------------------------------.
|                                                                           |
| WTOR Text... WHAT IS YOUR USER ID?                                        |
|                                                                           |
'---------------------------------------------------------------------------'
.- Response Text Analysis -------------------------------------------------.
|                                                                           |
|             Reply Match Text                                R/Code        |
|             USER01                                          10            |
|             USER02                                          20            |
|             USER03                                          30            |
|             USER04                                          40            |
|             USER05                                          50            |
|                                                                           |
|             Store Actual Reply Text in Variable Name  REPLYTEXT           |
|                                                                           |
|             Wait Time 120   Wait Time Expiry Return Code 32               |
'---------------------------------------------------------------------------'

 F1=Help      F2=Split     F3=Exit
                           F9=Swap
```

F

Message Text Analysis

Message text analysis enables you to make a message rule more specific by specifying search criteria relating to the actual text of a message.

This appendix describes the message text analysis feature.

Using Message Text Analysis

You specify the message text analysis criteria on the Define Extended Filter Definitions panel (for a resource definition), or the Message Validation panel (for a message action rule). The panels enable you to analyze the text of the received message on a word, phrase, or string basis by specifying a combination of values in the Strt Pos, Word Num, and Opr fields. You can specify up to five tests, which are then linked in a defined, logical relationship that you specify in the Expression field.

For example, the entry 1 AND (2 OR 3) in the Expression field means that test 1 *must* be true and *either* test 2 *or* test 3 *must* be true for the rule to be valid.

A message consists of words. A word is a string of characters delimited by either a *space* or a *comma*. You have the option of specifying a word or part of a word within a message for testing, or of extracting a substring of the message text for testing.

Caution

ResourceView handles numeric comparisons; EventView always compares character values.

Strt Pos

The value you enter in the Strt Pos field indicates a position within the message where the text comparison is to start. Valid values are 1 through 999. The default is 1. The actual starting position is determined by the presence or absence of a value in the Word Num field.

If the start position is 2 and the Word Num field is blank, the text used for the comparison is the partial message starting at the second character in the message.

If the Strt Pos field is blank, but there is a value specified in the Word Num field, then only that word is compared to the scan text. If both the Strt Pos *and* the Word Num field are blank, the entire message is compared to the scan text.

If entries are made in both the Strt Pos and Word Num fields, the comparison is narrowed to a start position within a single word of the message text and the text used for comparison is the partial word. For example, if the word number is 8 and the start position is 2, the text for the comparison starts from the second character of the eighth word.

For example, the following message arrives:

```
AAA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
```

- If the Strt Pos field has a value of 2, the string to be tested is as follows:

  ```
  AA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
  ```

- If the Word Num field has a value of 5, the string to be tested is as follows:

  ```
  MESSAGE
  ```

- If the Strt Pos field has a value of 2 and the Word Num field has a value of 5, the string to be tested is as follows:

  ```
  ESSAGE
  ```

Word Num

The value you enter in this field identifies a particular word in a specific position within the message text string. Valid values are blank and 1 through 999. If this field is blank, the entire message text that occurs after the specified start position is compared to the scan text. If this field contains a value but the Strt Pos field is blank, only the specified word is compared to the scan text.

Opr

If you enter a value in either the Strt Pos or the Word Num field, you must enter a valid operator to control the type of comparison to be performed. The following operators are valid: CT (ConTain), EQ (EQual to), GE (Greater than or Equal to), GT (Greater Than), LE (Less than or Equal to), LT (Less Than), and NE (Not Equal to). If you enter a question mark (?) in this field, the list of valid operators is displayed.

Scan Text

This field contains the actual text (scan text) you want to test against the message text. You must have a match in the specified position or word for the comparison to be true. If you have specified either CT or EQ as the operator, you can use the wildcard character within or at the end of the Scan Text field. (You cannot use the wildcard character with the other operators.)

Examples

The following subsections contain examples of the use of the different operators in the message text analysis process. Each example describes the type of operator, and shows:

- The message received
- The test criteria
- The actual text to be tested
- The result of the test

Note

Text checking is done using the EBCDIC codes.

Example of the Use of the CT Operator

The ConTain operator tests whether the extracted message text string (after the Strt Pos and Word Num fields have been applied) contains the specified scan text string anywhere within it. If the Strt Pos and Word Num fields are blank, then the comparison is true if the scan text string appears anywhere in the message text.

```
AAA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
```

Strt Pos	Word Num	Extracted Text	Scan Text	Result
3	5	SSAGE	AGE	TRUE
1	2	THIS	THIS	TRUE

Example of the Use of the EQ Operator

The EQual to operator tests for an exact match. That is, the (extracted) message text string must match the scan text exactly for the test to succeed.

The designated wildcard character can occur either within the scan text, where it represents a single variable character, or at the end of the scan text, where it represents any number of variable characters.

If, for example, the message text is FREDERICK and the scan text is FRED, the test fails. If, however, the scan text is FRED*, the test succeeds.

```
AAA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
```

Strt Pos	Word Num	Extracted Text	Scan Text	Result
4	5	SAGE	SAG	FALSE
4	5	SAGE	SAGE	TRUE
1	8	100	10	FALSE
1	8	100	100	TRUE

Example of the Use of the GE Operator

The Greater Than or Equal to operator tests whether the value of the (extracted) message text string is greater than or equal to that of the text string in the Scan Text field. For ResourceView, if the characters to be tested are numeric, a numeric comparison is executed.

```
AAA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
```

Strt Pos	Word Num	Extracted Text	Scan Text	Result
1	8	100	99	TRUE (ResourceView)
1	8	100	99	FALSE (EventView)
1	8	100	100	TRUE
4		100A THIS …	99	FALSE
4	5	SAGE	LAGE	TRUE
4	5	SAGE	TAGE	FALSE

Example of the Use of the GT Operator

The Greater Than operator tests whether the value of the (extracted) message text string is greater than that of the text string in the Scan Text field. For ResourceView, if the characters to be tested are numeric, a numeric comparison is executed.

```
AAA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
```

Strt Pos	Word Num	Extracted Text	Scan Text	Result
1	8	100	99	TRUE (ResourceView)
1	8	100	99	FALSE (EventView)
1	8	100	100	FALSE
4		100A THIS …	99	FALSE
4	5	SAGE	LAGE	TRUE
4	5	SAGE	TAGE	FALSE

Example of the Use of the LE Operator

The Less Than or Equal to operator tests whether the value of the (extracted) message text string is less than or equal to that of the text string in the Scan Text field. For ResourceView, if the characters to be tested are numeric, a numeric comparison is executed.

```
AAA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
```

Strt Pos	Word Num	Extracted Text	Scan Text	Result
1	8	100	99	FALSE (ResourceView)
1	8	100	99	TRUE (EventView)
1	8	100	100	TRUE
4		100A THIS …	99	TRUE
4	5	SAGE	LAGE	FALSE
4	5	SAGE	TAGE	TRUE

Example of the Use of the LT Operator

The Less Than operator tests whether the value of the (extracted) message text string is less than that of the text string in the Scan Text field. For ResourceView, if the characters to be tested are numeric, a numeric comparison is executed.

```
AAA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
```

Strt Pos	Word Num	Extracted Text	Scan Text	Result
1	8	100	99	FALSE (ResourceView)
1	8	100	99	TRUE (EventView)
1	8	100	100	FALSE
4		100A THIS …	99	TRUE
4	5	SAGE	LAGE	FALSE
4	5	SAGE	TAGE	TRUE

Example of the Use of the NE Operator

The Not Equal to operator tests for a mismatch between the (extracted) message text string and the text string in the Scan Text field.

```
AAA100A THIS IS A MESSAGE NUMBERED MESSAGE 100
```

Strt Pos	Word Num	Extracted Text	Scan Text	Result
1	8	100	99	TRUE
1	8	100	100	FALSE
4		100A THIS …	99	TRUE
4	5	SAGE	LAGE	TRUE
4	5	SAGE	TAGE	TRUE

Linking Tests in an Expression

The tests you specify in the Text Analysis box can be linked in a defined relationship in the Expression field, using the Boolean operators AND, OR, and NOT.

For example, if you specify `1 AND (2 OR 3)` in the Expression field, this indicates that test one must always be true, and either test two or test three must be true, before the rule can be triggered.

Note

If you leave the Expression field blank, all specified conditions must be true.

G

Boolean Operators

Boolean operators enable you to test whether a specified criteria is satisfied. You use the operators to form Boolean expressions that have values of either FALSE or TRUE (for example, in message text analysis).

Boolean operators consist of two types: relational and logical.

This appendix describes these operators. The applicable operators depend on what you are doing.

Relational Operators

A relational operator enables you to test if a specified relationship between two items is true. You use the operators to form Boolean relational expressions.

Following are the descriptions of the operators:

=, **E**, or **EQ**
: Tests whether a value equals a specified value.

p, ^, p=, ^=, **N**, or **NE**
: Tests whether a value is not equal to a specified value. Whether you use the p character or the ^ character depends on your terminal.

>, **G**, or **GT**
: Tests whether a value is greater than a specified value. If the tested items are numeric, a numeric comparison is executed.

>= or **GE**
: Tests whether a value is greater than or equal to a specified value. If the tested items are numeric, a numeric comparison is executed.

<, **L**, or **LT**
: Tests whether a value is less than a specified value. If the tested items are numeric, a numeric comparison is executed.

<= or **LE**
: Tests whether a value is less than or equal to a specified value. If the tested items are numeric, a numeric comparison is executed.

CT, **CON**, or **CONTAINS**
: Tests whether a value contains the specified value as a substring.

LIK or **LIKE**
: Tests whether a value matches the specified pattern. You use _ and % as the wildcard characters: _ represents a single character, and % represents zero or more characters.

Logical Operators

A Boolean logical operator enables you to combine two Boolean expressions.

Following are the descriptions of the operators:

AND

Tests whether two expressions are both true. The truth table is as follows:

AND		**Expression 2**	
		FALSE	**TRUE**
Expression 1	**FALSE**	FALSE	FALSE
	TRUE	FALSE	TRUE

NOT

Negates the truth value of a Boolean expression. The value of NOT FALSE is TRUE; the value of NOT TRUE is FALSE.

OR

Tests whether any of the expressions is true. The truth table is as follows:

OR		**Expression 2**	
		FALSE	**TRUE**
Expression 1	**FALSE**	FALSE	TRUE
	TRUE	TRUE	TRUE

H

Knowledge Base Definition Classes

ResourceView definitions created in the knowledge base are differentiated by classes.

This appendix identifies these classes. The classes that are available depend on the product you are using.

Definition Classes

Table H-1 lists the common definition classes. Classes that are specific to a particular product are identified in parentheses.

Table H-1. Knowledge Base Definition Classes

Definition	Class Name	Class Number
CA-XCOM manager (FT)	XCMGR	62
CA-XCOM monitor (FT)	XCMON	63
Channel Interface Processor (IP and NS)	CIP	44
CICS database (CICS)	CICDB	06
CICS file (CICS)	CICFIL	05
CICS link (CICS)	CICLNK	07
CICS transaction (CICS)	CICTRN	04
Communications storage manager (IP and NS)	CSM	48
CONNECT:Direct manager (FT)	CDMGR	08
CONNECT:Direct monitor (FT)	CDMON	09
CONNECT:Mailbox manager (FT)	CMMGR	27
CONNECT:Mailbox monitor (FT)	CMMON	28
Direct access storage device (MSP, VOS3, and OS/390)	DASD	16
Enterprise Extender (IP and NS)	EE	47
File transfer schedule monitor (FT)	FTSCHD	24
FTP manager (FT)	FTPMGR	25
FTP monitor (FT)	FTPMON	26
FTS manager (FT)	FTSMGR	22
FTS monitor (FT)	FTSMON	23
Initiator (MSP, VOS3, and OS/390)	INIT	10

(Sheet 1 of 2)

Table H-1. Knowledge Base Definition Classes

Definition	Class Name	Class Number
IP application (IP and NS)	ASMON	43
Job (MSP, VOS3, CICS and OS/390)	JOB	19
Job Entry Subsystem (MSP, VOS3, and OS/390)	JES	20
Line (MSP, VOS3, and OS/390)	LINE	14
Network control point monitor (SNA)	NCPMON	54
Open Systems Adapter (IP and NS)	OSA	46
Printer (MSP, VOS3, and OS/390)	PRT	11
Service	SVC	61
SNA group (NMA)	SNAGRP	03
Internal resource	INTNL	21
Router (IP and NS)	ROUTER	45
Spool (MSP, VOS3, and OS/390)	SPOOL	13
Started task (MSP, VOS3, CICS and OS/390)	STC	02
Sysplex resource (OS/390)	SYSPLX	29
Tape or cartridge unit (MSP, VOS3, and OS/390)	TAPE	18
TCP/IP stack (IP and NS)	STACK	51
User class resource	USRCLS	17

(Sheet 2 of 2)

Reporting

This appendix provides information about how to use the reporting facilities of Automation Services.

This appendix covers the following topics:

- About reporting
- Generating a report
- Searching for macros
- Searching for message rules
- Searching for processes
- Using your search output

About Reporting

The reporting facility allows you to search your Automation Services knowledge base by specifying certain criteria to retrieve any definitions that match those criteria. For example, you might want to make a change to a macro and you need to find out what other parts of your system are affected by the change.

You can also update the retrieved definitions. For example, you might want to change all references to a macro. You can search for the macro and retrieve the relevant definitions, including the processes that reference the macro. You can then update those processes to refer to the required macro.

There are two report formats that can be generated:

- List—provides a selection list of output in summary form
- Print—prints detailed information

Generating a Report

A report is generated by performing a search with specified criteria. To obtain a report, perform the following steps:

Step 1. Enter the **/ASADMIN.R** path. The Reporting menu is displayed (see Figure I-1).

Figure I-1. Reporting Menu

```
B   - Browse Macro, Message or Process Occurances
P   - Print Macro, Message or Process Reports
PS  - Print Statistics
```

Step 2. Enter one of the following at the Select Option ===> prompt to specify the type of report that you want:

- **B**—displays the output as a selection list with summary information from which you can browse, update, or copy.
- **P**—prints the output with detailed information.

A selection list of criteria is displayed (see Figure I-2).

Figure I-2. Criteria List for Report Searches

```
SOLVPROD------------------------- CAS : Lists --------------------------------3
Command ===>                                                  Scroll ===> CSR

                                                         S/=Select I=Information
    Description
    Search for Macro References
    Search for Message Rules
    Search for Process References
    **END**
```

Step 3. Enter **S** beside the search criteria that you require. Each criteria selection displays another panel for you to specify search parameters. See the following sections for a description of how to specify the search parameters for the criteria you have chosen.

Note

Enter **I** beside any of the listed criteria to get more information about the type of search it performs.

Searching for Macros

If you selected the Search for Macro References criteria, the Search for Macro panel is displayed (see Figure I-3).

Figure I-3. Search for Macro Panel

```
SOLVPROD----------- Automation Services : Search for Macro --------------------
Command ===>                                                   Function=Search


 Macro Name .......... ________  (Name or Mask)

 NCL Procedure ....... ________  (Name or Mask)

 Macro Description ... ___________________________________  (Contains)


_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
```

The macro search allows you to retrieve macros that satisfy specified criteria. To search for macros, complete the following steps:

Step 1. Specify one or more search parameters using the fields provided on the Search for Macro panel. The following fields are provided:

Macro Name	Specifies a macro name or a mask. For example, *VAR* specifies macro names that contain VAR.
NCL Procedure	Specifies the name of an NCL procedure or a mask. For example, $RMMC2* specifies NCL procedure names that begin with $RMMC2.
Macro Description	Specifies a string of text to search for in macro descriptions.

Step 2. Press F6 (Action) to begin the search. The output is printed or listed depending on what type of report you specified on the Reporting Menu.

An example list output is shown in Figure I-4 on page I-5. The search criteria are as follows:

Field	**Value**
Macro Name	S*
Macro Description	process

The example retrieves all macros with names that begin with S and contain the word, *process*, in their descriptions.

Note

Press F11 (Right) to view additional information not displayed on a screen.

Figure I-4. Macro Search Result List

```
SOLVPROD--------- Automation Services : Search Results ---------------------1/5
Command ===>                                                   Scroll ===> CSR

                                                     S/B=Browse U=Update C=Copy
    System          Object
    Name      Vers  Name               Description
    $$MAC     0000  SETVARS            Create variables for a Process
    $$MAC     0000  SMFWRITE           Perform an SMFWRITE on behalf of a Proc
    $$MAC     0000  SNMPTRAP           Issue an SNMPTRAP on behalf of a Proces
    $$MAC     0000  SUBPRCSS           Submit a Process to a Background Region
    $$MAC     0000  SVAPI              Allow access to ServiceView API from a
    **END**
```

```
SOLVPROD--------- Automation Services : Search Results ---------------------1/5
Command ===>                                                   Scroll ===> CSR

                                                     S/B=Browse U=Update C=Copy
    System          Object
    Name      Vers  Name               Class
    $$MAC     0000  SETVARS            70 - Macro
    $$MAC     0000  SMFWRITE           70 - Macro
    $$MAC     0000  SNMPTRAP           70 - Macro
    $$MAC     0000  SUBPRCSS           70 - Macro
    $$MAC     0000  SVAPI              70 - Macro
    **END**
```

```
SOLVPROD--------- Automation Services : Search Results ---------------------1/5
Command ===>                                                   Scroll ===> CSR

                                                     S/B=Browse U=Update C=Copy
    System          Object             Created          Created   Created
    Name      Vers  Name               Date             Time      Userid
    $$MAC     0000  SETVARS            WED 30-JUL-1997  00.00.00  SOV3.1.3
    $$MAC     0000  SMFWRITE           WED 30-JUL-1997  00.00.00  SOV3.1.3
    $$MAC     0000  SNMPTRAP           WED 30-JUL-1997  00.00.00  SOV3.1.3
    $$MAC     0000  SUBPRCSS           WED 30-JUL-1997  00.00.00  SOV3.1.3
    $$MAC     0000  SVAPI              WED 30-JUL-1997  00.00.00  SOV3.1.3
    **END**
```

```
SOLVPROD--------- Automation Services : Search Results ---------------------1/5
Command ===>                                                   Scroll ===> CSR

                                                     S/B=Browse U=Update C=Copy
    System          Object             Updated          Updated   Updated
    Name      Vers  Name               Date             Time      Userid
    $$MAC     0000  SETVARS            WED 30-JUL-1997  00.00.00  SOV3.1.3
    $$MAC     0000  SMFWRITE           WED 30-JUL-1997  00.00.00  SOV3.1.3
    $$MAC     0000  SNMPTRAP           WED 30-JUL-1997  00.00.00  SOV3.1.3
    $$MAC     0000  SUBPRCSS           WED 30-JUL-1997  00.00.00  SOV3.1.3
    $$MAC     0000  SVAPI              WED 30-JUL-1997  00.00.00  SOV3.1.3
    **END**
```

The following information is displayed for each entry:

- System Name
- System Version
- Object Name
- Description
- Class (macro)
- Date and time the object was created
- ID of the user that created the object
- Date and time the object was last updated
- ID of the user that last updated the object

Searching for Message Rules

If you selected the Search for Message Rules criteria, the Search for Message Rule panel is displayed (see Figure I-5).

Figure I-5. Search for Message Rule Panel

```
SOLVPROD--------- Automation Services : Search for Message Rule ---------------
Command ===>                                                     Function=Search

Ruleset .......+ ________              Rule Status ............+ ________
Rule ID ........ ____________          Message ID(of major line) ____________
Message Text ... ____________________________________________________________

Job Name ....... ________              Rule Priority ........... __
Job Type ......+ _______               Execute If Not Best Fit?  ___
System Name .... ________

Deliver ........ ________
Replacement Text ___________________________________________________________
                 ___________________________________________________________

Reply Text ..... ___________________________________________________________
                 ___________________________________________________________
System Command . ___________________________________________________________
                 ___________________________________________________________
MS Command ..... ___________________________________________________________
                 ___________________________________________________________
Process .......+ ________              Command .......+ ________
 F1=Help      F2=Split     F3=Exit                                F6=Action
                           F9=Swap
```

The message rule search allows you to locate all message rules that match the specified criteria. To search for message rules, complete the following steps:

Step 1. Specify one or more search parameters using the fields provided on the Search for Message Rule panel. The following fields, which are a subset of the fields that are displayed when you define message rules, are provided:

Ruleset	specifies a ruleset name.
Rule ID	specifies the first word of the message text.
Rule Status	specifies the status of a rule.
Message ID	specifies the first word of a message major line.
Message Text	specifies the text to compare with the received messages.
Job Name	specifies the name of the job that sends the message.
Job Type	specifies the type of the job that sends the message.
Rule Priority	specifies the priority assigned to the rule.
Execute If Not Best Fit?	specifies the rule best fit execution criteria.
System Name	specifies the system from which a message originates.
Deliver	specifies the message delivery destinations.
Replacement Text	specifies the text to replace a message.
Reply Text	specifies the text in reply to a WTOR message.
System Command	specifies the system command to issue when a rule is triggered.
MS Command	specifies the Management Services command to issue when a rule is triggered.
Process	specifies the process to execute when a rule is triggered.
Command	specifies the Automation Services command to issue when a rule is triggered.

These fields do not support masks. For example, using IEC03* as the rule ID does not retrieve all rules with IDs that begin with IEC03. Instead, the criterion retrieves only those rules with message texts that have a first word of IEC03*. The asterisk (*) is interpreted as a real character, not a wild card.

Step 2. Press F6 (Action) to begin the search. The output is printed or listed depending on what type of report you specified on the Reporting Menu.

An example list output is shown in Figure I-6. The search criterion is that the rule set name be AGRSUPP. The example retrieves all message rules in the AGRSUPP rule set.

Figure I-6. Message Rule Search Result List

```
SOLVPROD----------- Automation Services : Search Results -----------------1/128
Command ===>                                                   Scroll ===> PAGE

                                                     S/B=Browse U=Update C=Copy
    System           Object
    Name      Vers   Name          Description
    AGRSUPP   ZEVW   QAN77284506950345  R, TEXT VIA $DM
    AGRSUPP   ZEVW   QAN77284506950398  AUTOMATIC COMMANDS HALTED
    AGRSUPP   ZEVW   QAN77284506950449  I/O ERRORS ON LINE XX
    AGRSUPP   ZEVW   QAN77284506950505  ON TSORDR/INTRDR/ETC....RDR
    AGRSUPP   ZEVW   QAN77284506950554  JOBNAME HELD
    AGRSUPP   ZEVW   QAN77284506950602  JOBNAME -- ILLEGAL JOB CARD
    AGRSUPP   ZEVW   QAN77284506950648  JOBNAME -- INVALID /*ROUTE CARD
    AGRSUPP   ZEVW   QAN77284506950698  JOBNAME -- INVALID /*JOBPARM CARD
    AGRSUPP   ZEVW   QAN77284506950749  JOBNAME -- INVALID /*OUTPUT CARD
    AGRSUPP   ZEVW   QAN77284506950804  JOBNAME -- INVALID EXECUTION MODE
    AGRSUPP   ZEVW   QAN77284506950853  JOBNAME -- INVALID /*NETACCT CARD
    AGRSUPP   ZEVW   QAN77284506950934  JOBNAME -- INVALID /*NOTIFY CARD
    AGRSUPP   ZEVW   QAN77284506950992  JOBNAME -- INVALID /*XMIT CARD
    AGRSUPP   ZEVW   QAN77284506951044  JOBNAME -- INVALID /*CONTROL STATEMENT
    AGRSUPP   ZEVW   QAN77284506951087  DEVNAME COMMAND (eg. INTRDR $VS, 'CMD')
    AGRSUPP   ZEVW   QAN77284506951132  SKIPPING FOR JOBCARD
 F1=Help      F2=Split     F3=Exit                  F5=Find      F6=Refresh
 F7=Backward  F8=Forward   F9=Swap                 F11=Right
```

The following information is displayed for each entry:

- System Name (name of the rule set)
- System Version (reserved EventView version)
- Object Name (internal message rule ID)
- Description
- Class
- Date and time the object was created
- ID of the user that created the object
- Date and time the object was last updated
- ID of the user that last updated the object

Note

Press F11 (Right) to view additional information not displayed on the screen.

Searching for Processes

If you selected the Search for Process References criteria, the Search for Process panel is displayed (see Figure I-7).

Figure I-7. Search for Process Panel

```
SOLVPROD---------- Automation Services : Search for Process -------------------
Command ===>                                                   Function=Search


 System Name ......+ ________    Version ......+ ____

 Process Name ...... ________    (Name or Mask)

 Process Description ... ________________________________  (Contains)
```

The process search allows you to locate all processes that match the specified criteria and the definitions in the knowledge base where the processes are referenced. Processes might be referenced in availability maps, resource definitions, or message rules.

To search for processes, complete the following steps:

Step 1. Specify one or more search parameters using the fields on the Search for Process panel. The following fields are provided:

System Name	Specifies the name of a system image.
Version	Specifies the version of the system image.
Process Name	Specifies the name of a process or a mask. For example, DISP* specifies process names that begin with DISP.
Process Description	Specifies a string of text to search for in process descriptions.

Note

It is recommended that you specify a system name and version when performing the process search.

Step 2. Press F6 (Action) to begin the search. The output is listed or printed depending on the type of report you specified on the Reporting Menu.

An example list output is displayed in Figure I-8. The search criteria are as follows:

Field	Value
System Name	$TEMPLAT
Version	1
Process Name	DISP*

The example retrieves all processes with names that begin with DISP in the $TEMPLAT 0001 system image, and all definitions that use those processes.

Figure I-8. Process Search Result List

```
SOLVPROD--------- Automation Services : Search Results --------------------1/51
Command ===>                                                   Scroll ===> CSR

                                                     S/B=Browse U=Update C=Copy
    System          Object
    Name      Vers  Name              Description
    $TEMPLAT  0001  JES2INIT          Initiator Template Definition
    $TEMPLAT  0001  JES2PRT           Printer Test Template Local
    $TEMPLAT  0001  STDCART           STANDARD TAPE TEMPLATE
    $TEMPLAT  0001  STDTAPE           STANDARD TAPE TEMPLATE
    $TEMPLAT  0001  DISPINIT          Display an INITIATOR and Set State (Sup
    $TEMPLAT  0001  DISPPRTR          Display a PRINTER and Set State    (Sup
    $TEMPLAT  0001  DISPTAPE          Display a TAPE and Set State       (Sup
    $TEMPLAT  0002  STDCART           STANDARD TAPE TEMPLATE
    $TEMPLAT  0002  STDTAPE           STANDARD TAPE TEMPLATE
    $TEMPLAT  0666  JES2INIT          Initiator Template Definition
    $TEMPLAT  0666  JES2PRT           Printer Test Template Local
    $TEMPLAT  0666  STDCART           STANDARD TAPE TEMPLATE
    $TEMPLAT  0666  STDTAPE           STANDARD TAPE TEMPLATE
```

The following information is displayed for each entry:

- System Name
- System Version
- Object Name
- Description
- Class (process or referencing definition)
- Date and time the object was created
- ID of the user that created the object
- Date and time the object was last updated
- ID of the user that last updated the object

Note

Press F11 (Right) to view additional information not displayed on the screen.

Using Your Search Output

If you have chosen to list the output of your search you can browse, update, or copy any entry in the selection list. You can also issue the SORT and LOCATE commands to help you view and find information in the list.

If you have chosen to print your search output, you can hold the output and view it online before printing. See the *Management Services User's Guide* for information on how to use the print services manager.

Sorting Your Output

To sort the output in the selection list perform the following steps:

Step 1. Enter **SORT ?** at the Command ===> prompt. A list of sort criteria is displayed (see Figure I-9).

Figure I-9. Sort Criteria Panel

```
SOLVPROD----------------- NCL/EF : NDB Field List --------------------------14
Select Option ===>                                            Scroll ===> PAGE

          Field Name    Description
   1    - $RM00CDATE    RESOURCE CREATION DATE
   2    - $RM00CLASS    OBJECT CLASS GROUPING
   3    - $RM00LMDATE   RESOURCE LAST MODIFIED DATE
   4    - $RM00LMTIME   RESOURCE LAST MODIFIED TIME
   5    - $RM00NAME     RESOURCE NAME
   6    - $RM00SDESC    RESOURCE DESCRIPTION
   7    - $RM00SYSNAME  SYSTEM NAME
   8    - $RM00VERSION  DEFINITION VERSION
          **END**
```

Step 2. Enter the number of the sort criterion you require at the Select Option ===> prompt. You are returned to your search selection list with the SORT command and the criteria displayed at the command prompt.

Step 3. Press ENTER to action the sort.

Locating Output

You can use the LOCATE command to find specific entries in your search selection list. The command is performed on the field on which you sorted.

For example, if you wanted to find the process $PROC01 in your search selection list, proceed as follows:

Step 1. Enter **SORT $RM00NAME** at the Command ===> prompt to sort the list by the object name.

Step 2. Enter **L $PROC01** to locate the specified process.

J

Message Routing Codes

One or more routing codes are usually associated with a system message. The operating system uses routing codes to determine which messages are to be delivered to a particular console. When a system console is generated, it is assigned one or more routing codes. In Automation Services, routing codes can be used to determine which messages are to be sent to a consolidated console.

This appendix describes these message routing codes.

Message Routing Codes

The operating system, by convention, applies certain meanings to specific routing codes. You are encouraged to work within these conventions. System-sourced messages will follow these conventions.

Note

MSP and VOS3 systems support routing codes 1 through 16 only.

The following values for message routing codes are defined:

1	Master console action
2	Master console information
3	Tape pool
4	Direct access pool
5	Tape library
6	Disk library
7	Unit record pool
8	Teleprocessing control
9	System security
10	System error/maintenance/system programmer information
11	Programmer information
12	Emulators
13	Reserved for customer use
14	Reserved for customer use
15	Reserved for customer use
16	Reserved for customer use
17-20	Reserved for customer use
21-28	Reserved for subsystem use
29-41	Reserved for IBM use
42	JES2/JES3 information
43-64	Reserved for JES use only
65-96	Processor related messages
97-128	Device-related messages

K

Message Descriptor Codes

One or more descriptor codes are usually associated with a system message. The operating system uses descriptor codes to determine the display attributes of a message. Descriptor codes 1, 2, and 11 indicate that the message is non-roll deletable (NRD). Descriptor code 7 indicates NRD if the message is also a WTOR.

Descriptor codes 1 through 6 and 11 are mutually exclusive, whereas codes 7 through 10 can be used in combination with any other code.

This appendix describes these message descriptor codes.

Message Descriptor Codes

The following values for message descriptor codes are supported:

1	System failure
2	Immediate action required
3	Eventual action required
4	System status
5	Immediate command response
6	Job status
7	Application program/processor
8	Out-of-line message
9	Operator request
10	Dynamic status displays
11	Critical eventual action requested
12	Reserved for future use
13	Reserved for future use
14	Reserved for future use
15	Reserved for future use
16	Reserved for future use

L

Message Levels

Message levels can be used to limit the system messages that can be delivered to a specific system console or consolidated console user.

When a user ID is defined with authority to receive messages captured by the Automation Services subsystem interface, the messages to be delivered to that user can be limited by specifying one or more values for message level.

This appendix describes these message levels.

Message Levels

The following values for message levels are supported:

WTOR
: write-to-operator with reply messages, which usually require an operator reply.

I
: system failure and immediate action messages (descriptor codes 1 and 2), which indicate a task is waiting for operator action.

CE
: critical eventual action messages (descriptor code 11), which indicate a potential system problem.

E
: eventual action messages (descriptor code 3), which do not require immediate action.

IN
: informational messages.

BC
: console broadcast messages.

Glossary

ACB (access method control block)

The ACB is a control block that links an application program to an access method such as IBM's VTAM.

Activity Log

The activity log is a log of all important activities occurring in a region. You can use the log to assist you in determining the cause of a problem.

Actual State

The actual state is the state a defined resource is actually in at any given time. In the AUTOMATED mode, ResourceView keeps the actual state consistent with the desired state.

The valid actual states are: ACTIVE, DEGRADED, FAILED, INACTIVE, RECOVERED, STARTING, STOPPING, and UNKNOWN.

AIM (advanced information manager)

The advanced information manager is an integrated online database system that interfaces with the MSP operating system.

Alert

An alert is an event displayed on the alert monitor. An alert can be internally generated (for example, when a monitored resource fails) or user-defined.

AOM (Advanced Operation Management)

The AOM is a facility that manages and controls local and remote operating systems. This is the subsystem interface to a region supported by Automation Services. The interface enables the passage of system messages, events, and commands.

APF (authorized program facility)

The authorized program facility identifies programs that are authorized to use restricted system functions.

API (application program interface)

The application program interface is a functional interface supplied by a program. The interface enables another application, written in a high-level language, to use specific data or functions of the program.

APPC (advanced program-to-program communications)

APPC is an IBM-defined application-level protocol (which makes use of SNA's LU 6.2) that allows interconnected systems to communicate and share the processing of programs.

AUTOMATED Mode

The AUTOMATED mode is a service and resource operation mode that allows you to automate many of the functions previously done by operators at consoles. Other modes of operation are OFF, MANUAL, and IGNORED.

Availability Map

When a region is implemented, each resource or service defined in the knowledge base can be attached to an availability map. This map indicates the time that the resource or service is available to the system.

BDAM (basic direct access method)

The basic direct access method is used to retrieve or update particular blocks of a dataset on a direct access device.

Boolean

Boolean is an adjective that describes a logical entity, operator, operation, or expression, and is named after the philosopher George Boole.

Boolean Expression

A Boolean expression is a logical expression that evaluates to either false or true.

CDRM (cross-domain resource manager)

The cross-domain resource manager provides the SSCP with descriptions and addresses for all the SNA resources available in other domains.

Child

A child is an entity that has one or more parents. A child *cannot* start in an automated startup sequence until all the parents have become active.

CICS (Customer Information Control System)

The Customer Information Control System is an IBM-licensed program that enables transactions entered at remote terminals to be processed concurrently by user-written application programs. It also includes facilities for building, using, and maintaining databases.

CNM (communication network management) interface

The communication network management interface provides an access method to application programs for handling data and commands associated with communication system management.

Consolidated Console

The consolidated console enables you to view messages from linked operation regions at a single monitor.

The link between regions occurs at the event level, where INMC enables one region to ship messages to another region.

CPU (central processing unit)

The central processing unit is the part of a computer that includes the circuits that control the interpretation and execution of instructions.

Cross-domain Resource

A cross-domain resource is a network resource that is owned by the VTAM in another domain.

CSD (CICS system definition)

The CICS system definition is a dataset that contains a resource definition record for every record defined to CICS.

DAE (dump analysis and elimination)

Dump analysis and elimination is an OS/390 process that suppresses duplicate dumps.

DASD (direct access storage device)

A direct access storage device is a storage device in which the access time is effectively independent of the location of the data (for example, a disk drive).

DBCTL (Database Control)

Database Control is an interface between CICS/ESA and IMS/ESA that allows access to IMS DL/I databases from one or more CICS regions without the need for data sharing. It provides release independence, virtual storage constraint relief, operational flexibility, and failure isolation.

DCT (destination control table)

The destination control table contains entries for the transient data queues used by a CICS region.

Desired State

The desired state is the required state of a resource. Desired state can be overridden by an availability map or manually from a monitor.

The valid values are ACTIVE and INACTIVE.

Display Attribute Tables

The display attribute tables contain display attribute values for resources and icons on the status and graphical monitors. The logical state of a resource determines its display attributes. Authorized users can customize display attributes.

DL/I (Data Language I)

The DL/I is a high-level language interface between an application program and the Information Management System (IMS). The interface provides for data definition, manipulation, and control.

DOM (delete operator message)

A delete operator message enables the deletion of a non-roll delete message (NRD) from the message monitor or console.

Domain

A domain consists of the set of SNA resources controlled by an SSCP. In terms of implementation, an SSCP is the host access method (VTAM). An SNA network consists of one or more domains.

EBCDIC (extended binary-coded decimal interchange code)

The extended binary-coded decimal interchange code provides a coded character set consisting of 8-bit coded characters.

ENF (event notification facility)

The event notification facility is an OS/390 component that allows authorized programs to listen for specific system events.

EPS (EndPoint Services)

EndPoint Services is a communications access method provided by the subsystem interface (SSI) regions. The method provides communication by using the OS/390 cross-system coupling facility (XCF) in a sysplex environment.

ESA (Enterprise Systems Architecture)

Enterprise Systems Architecture defines the conceptual structure and functional behavior of a range of IBM mainframe computers.

ESF (Expert Systems Foundation)

ESF is a rule-based facility that automates the handling of messages captured by AOM.

Exit

An exit is a user-written program or procedure that can be invoked by a region. The exit provides data to the region and performs actions that the region does *not* directly support but that are required by the installation.

Graphical Monitor

The graphical monitor is a facility that enables you to monitor and control resources. The monitor provides an iconic view of groups of resources.

Heartbeat Feature

The heartbeat feature enables you to specify a time interval at which the region checks the status of the resource. You use the feature to monitor resources that are liable to change state without an accompanying message.

IBM TCP/IP

IBM TCP/IP refers to any IBM TCP/IP stack, including SecureWay CS, eNetwork CS, and TCP/IP for MVS.

Icon

An icon is a graphical representation of grouped services and resources (for example, all printers in a system image) that is displayed on the graphical monitor. If the icon changes color, an operator can check the resources in the resource group to see what has happened and why.

Icon Panel

An icon panel is a predefined panel that contains icons that represent groups of resources. For example, an icon panel might contain an icon for the printers in a system, another icon for the started tasks in a system, and so on.

ICS (Initialization and Customization Services)

The Initialization and Customization Services is an AutoAssist facility that helps you set up your region parameters.

IDCM (Integrated Data Communication Manager)

The Integrated Data Communication Manager is the base product for communication control in the AIM database system.

IMS (Information Management System)

The IMS is a database management system that enhances an operating system to provide database processing in both batch and online environments.

INMC (Inter-Management Services Connection)

The INMC is a Management Services component that allows multiple regions to be linked and controlled from a single location.

Internet

The internet is a collection of linked networks using TCP/IP and related protocols.

IRC (interregion communication)

Interregion communication is the method by which CICS provides communication between a CICS region and another region on the same system. The method is used by MRO.

IS (information system)

The information system is a computer system that provides information according to user requests.

ISC (intersystem communication)

Intersystem communication is communication between separate systems by means of SNA networking facilities or VTAM application-to-application facilities. It links CICS regions and other regions, and is used for communication between user applications or for executing CICS functions in a remote CICS region.

ISR (Inter-System Routing)

The ISR feature enables the routing of information between regions.

JCL (job control language)

The job control language is used to identify a job to an operating system and to describe the job requirements.

JES (job entry subsystem)

The job entry subsystem is a 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.

Knowledge Base

A knowledge base is a database in which you store the policies and procedures that govern the operation of your systems.

Link Station

In VTAM, a link station is a named resource within a subarea node that represents another subarea node that is attached by a subarea link. In the resource hierarchy, the link station is subordinate to the subarea link.

Logical Database

Every region has its own physical database; but in a multisystem environment, you see databases in the linked regions as one logical database.

The changes made to a definition held in a database are reflected across the databases in all regions, regardless of their physical location.

Logical Resource

Some resources function purely to initiate some sort of activity from another resource. For example, you can define a resource purely for the purpose of starting or stopping other resources. Other resources might be logical representations of groups of resources. These types of resources need *not* be physical resources. Define these resources as LOGICAL resource types.

Logical State

The logical state is the state of a resource based on its actual state, desired state, and operation mode. The logical state of a resource determines its display attributes, and alerts the operator to changes in resource state. The operator can then take action if necessary.

The valid logical states are: OK, ATTENTION, INERROR, PENDING, STARTING, STOPPING, FAILED, DEGRADED, and UNKNOWN.

LSR (local shared resource)

Local shared resource pools are used for buffering I/O to VSAM files. NCL supports this type of processing for user databases (UDBs).

LU (logical unit)

SNA introduced the concept of the logical unit (LU). The LU is a type of SNA network addressable unit (NAU) that enables end users to gain access to network resources and communicate with each other.

LU 0

LU 0 is an unconstrained SNA protocol that allows the selection of any set of available protocol rules, as long as the two LUs are able to communicate with each other successfully according to the rules chosen. Therefore, all LU types are an implementation of LU 0.

LU 1

LU1 is an SNA protocol that is used by line-by-line or typewriter type terminals (for example 3767 and 3770).

LU 6.2

LU 6.2 is a protocol that serves as a port into an SNA network. LU 6.2 defines a specific set of services, protocols, and formats for communication between logical processors. LU 6.2 provides presentation services for presentation of data to the end user, transaction services for performing transaction processing on behalf of the end user, and LU services for managing the resources of the LU.

Macro

See Registered Macro.

Major Node

A major node is a collection of network resources that can be activated or deactivated as a group.

MCS (multiple console support)

The MCS provides the operator interface in an OS/390 system.

MDO (Mapped Data Object)

A Mapped Data Object is any data item that can be represented as a continuous string a bytes in storage.

Message Profile

A message profile is a set of criteria that determines which messages are displayed on a message consolidation console.

MODS (Managed Object Development Services)

The Managed Object Development Services is a development environment that provides tools for creating and customizing NCL applications.

MRO (multiregion operation)

Multiregion operation provides communication between CICS regions in the same processor without the use of SNA network facilities. It enables these regions to communicate with each other, and to share resources such as files, terminals, and temporary storage.

MSP (Multidimensional System Products)

MSP is a Fujitsu operating system.

MVS (Multiple Virtual Storage)

MVS is an IBM operating system.

NAU (network addressable unit)

In SNA, a network addressable unit can be a logical unit, a physical unit, or a system services control point. Each unit has a unique network address that identifies it to other nodes in the network.

NCL (Network Control Language)

Network Control Language is a structured interpretive language available in a region. The language enables you to develop logical procedures (programs) for performing specific tasks. NCL contains a wide range of logic, built-in functions, and arithmetic facilities which can be used to provide powerful monitoring and automatic control functions.

NCL Procedure

An NCL procedure is a member of the Management Services procedures dataset comprising NCL statements. The NCL statements are executed by an EXEC or a START command specifying the name of the procedure.

NCL Process

An NCL process is the NCL task that is invoked by an EXEC or a START command to execute one or more NCL procedures. Each NCL process has a unique NCL process identifier.

NCP (Network Control Program)

This program resides within and controls the operation of a communications controller. The NCP communicates with VTAM.

NDB

An NDB is a database in VSAM Key Sequenced Dataset (KSDS) format.

Node Type

In SNA, node type designates a node according to the protocols it supports and the network addressable units (NAUs) it can contain. Five types are defined: 1, 2.0, 2.1, 4, and 5. Type 1, type 2.0, and type 2.1 nodes are peripheral nodes; type 4 and type 5 nodes are subarea nodes.

NPF (Network Partitioning Facility)

Network Partitioning Facility is a Management Services facility that defines the functions a user can perform.

Non-roll Delete (NRD) Message

A non-roll delete message is a message on an OCS screen. The message indicates an event of critical importance and does *not* roll off the screen as the screen fills up (for example, a message that requires a reply). You remove an NRD message by deleting the message explicitly or by replying to the message.

NST (non-SNA terminals)

NST is a feature that enables non-VTAM terminal support.

OCS (Operator Console Services)

The Operator Console Services is a Management Services facility that provides a central point of message monitoring and operational control either locally or for a network of regions.

Packet

A packet is a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole. The data, control signals and, possibly, error control information are arranged in a specific format.

Parameter Group

A parameter group contains parameters that determine the characteristics of a region.

Parent

A parent is an entity that has one or more children. A parent *cannot* be stopped in an automated shutdown sequence until all the children have become inactive.

Peripheral Node

A peripheral node provides end user access to an SNA network.

Physical Database

Every region has its own physical database. The database stores, for example, system image definitions, resource definitions, process definitions, and rule set definitions for that region.

The changes made to a definition held in a database are reflected across the databases in all connected regions, regardless of their physical location.

The fact that each region has its own database means that the region can run in isolation if the links to other regions fail.

PLT (program list table)

The program list table is a CICS control table that contains a list of programs. The programs in a PLT can be executed as a group during CICS startup or shutdown, and can be enabled and disabled as a group by a single CEMT transaction.

PPI (program-to-program interface)

The program-to-program interface enables programs to exchange data.

PPO (primary program operator)

The primary program operator is a facility of VTAM that allows unsolicited network messages to be delivered to an application program for processing.

Process

A process is a means of automating a series of commands and actions. For example, the sequence of actions involved in shutting down CICS can be translated into a series of steps in a single process. A macro performs the actions required by each step.

Prompted Field

A prompted field is a field that is linked to a list of values. A user can select one of these values to complete the field. Authorized users can edit the lists.

PSM (Print Services Manager)

The Print Services Manager enables you to control the physical printing of reports on JES or network printers.

PU (physical unit)

Each node (a logical grouping of hardware) in an SNA network is addressed by its PU.

PUT (program update tape)

The program update tape is a tape containing software updates.

RACF (Resource Access Control Facility)

The Resource Access Control Facility is an IBM-licensed program that provides for access control by identifying users to the system and verifying their authority, authorizing access to DASD data sets, logging detected unauthorized attempts to enter the system, and logging detected accesses to protected data sets.

Registered Command

A registered command contains instructions to the region about the actions to take.

Registered Macro

A registered macro points to an NCL procedure that performs the actions in a process step.

Relationships

There are two types of relationships: parent and child. A relationship determines, for example, the automated startup and shutdown sequences of resources. A parent must be active before a child can be started. A child must be inactive before a parent can be stopped. A parent can have more than one child, and a child can have more than one parent.

Resource

A resource is an entity used to provide a function or, together with other resources, to provide a service (for example, a started task, a printer, or an SNA application).

Resources are defined to a region as part of a system image. A resource definition contains the operations policies and methods for a resource.

Resource Group

A resource group represents a group of resources. You attach resource groups to icons. An operator can issue a command to look at the resources if, for example, an icon changes color. The icon might have an attached resource group that contains, for example, all the printers in a system image.

Resource Group Filter

A resource group filter is a set of instructions that determines which resources are contained in a resource group.

Resource States

A resource defined to a region has three types of states: desired, actual, and logical. The region aims to maintain the resource at the desired state.

The logical state indicates the condition of a resource, based on its desired state, actual state, and operation mode. The logical state determines the display attributes of a resource and alerts the operator to any problems.

Resource Template

A resource template provides predefined values for the definition of a commonly-used resource.

Templates for different resources can be copied to different systems and system images.

SDO (service-driven operations)

The SDO is an operations approach that views the information system from an enterprise-wide perspective and manages the system in terms of the services it delivers to the business.

SMF (system management facility)

The system management facility is a standard feature of MSP and OS/390 that provides the means for gathering and recording information that can be used to evaluate system usage.

SMS (System Management Service)

The System Management Service is a VOS3 feature that provides the means for gathering and recording information that can be used to evaluate system usage.

SNA (Systems Network Architecture)

The Systems Network Architecture is a set of standards that allows the integration of all the different mainframe hardware and software products into a universal network. The standards describe the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks.

SNMP (Simple Network Management Protocol)

SNMP is a protocol for the management of the internet.

SOLVE

SOLVE encompasses the services provided by Management Services and Automation Services. For example, the SOLVE PPI is a service provided by the subsystem interface (SSI) in Management Services.

SSCP (system service control point)

A system service control point is the SNA element that provides the network management and control functions for a network or a portion of a network.

Status Monitor

The status monitor enables you to monitor and control individual resources. The monitor displays resource statuses in line-by-line mode. These statuses are color-coded to alert an operator to changes in resource status. Changes in color are governed by changes to the *logical state* of a resource.

Status Monitor Filter

A status monitor filter is a Boolean expression that determines which resources are to be displayed on the status monitor. For example, an operator may use a filter that only displays the printers in a system image.

Subarea

A subarea is a set of resources controlled by a subarea node.

Subarea Node

A subarea node routes data through an SNA network. Type 4 and type 5 nodes are subarea nodes.

System Image

A system image is part of the knowledge base and identifies the resources that are to be managed:

- A local image contains local resources that are managed by the local region. Each managed system has its own image.
- A shared image contains resources that may be managed by different regions. A resource in a shared image may move from system to system.
- A sysplex image contains only sysplex resources such as the automatic restart manager (ARM).

TCP/IP (Transmission Control Protocol/Internet Protocol)

The Internet Protocol routes packets across multiple subnetworks (such as LANs and WANs). The Transmission Control Protocol is used on top of IP and provides guaranteed delivery of data across the internet.

Transient Log

The transient log is a log of activities associated with a resource that is monitored. One transient log exists for each resource definition loaded in a region and exists as long as the definition remains loaded in the region. The resource definition can, however, direct the log entries to the activity log for storage.

TSO (Time Sharing Option)

The Time Sharing Option allows terminal operators to interact directly with computer resources and facilities.

UAMS (User Access Maintenance Subsystem)

User Access Maintenance Subsystem is the security component of Management Services that enables the definition of users and the associated user authority.

UDP (User Datagram Protocol)

User Datagram Protocol is a packet-level protocol used on top of IP. It enables multiplexing of IP datagrams between different application processes but does not guarantee delivery. It is used by SNMP.

Variable

A variable is used to store data that can change. A variable is represented by a word that starts with an ampersand (&), followed by the name of the variable. For example, &A is a variable where A is the name. When &A is processed, it is replaced by the stored value.

VOS3 (Virtual-storage Operating System 3)

VOS3 is a Hitachi operating system.

VSAM (Virtual Storage Access Method)

The Virtual Storage Access Method is used for processing data files that utilize relative, sequential, and addressed access techniques.

VTAM (Virtual Telecommunications Access Method)

The Virtual Telecommunications Access Method is an IBM-licensed program that controls the communication between terminals and application programs in an SNA network.

WTO (write-to-operator)

WTO is a user-coded service that enables a message to be written to the system console operator, to inform the operator of errors and unusual system conditions that need correcting.

WTOR (write-to-operator with reply)

WTOR is a user-coded service that enables a message to be written to the system console operator, to inform the operator of errors and unusual system conditions that need correcting, and to which a response is expected.

XCF (cross-system coupling facility)

The cross-system coupling facility is an OS/390 component that provides functions to support cooperation between authorized programs running in a Sysplex environment.

XRF (extended recovery facility)

The extended recovery facility is an ESA function that enables you to minimize the effect of failures on the users of a software product.

Index

Symbols

##RESET##, 5-40
$AA- message prefix, 5-35
$DN- message prefix, 5-35
$LOG process parameter, 8-16
$MN- message prefix, 5-35
$NCL process, 5-42
$RMCLOG NCL procedure, B-5
$RMCONS NCL procedure, B-2
$RMEXSHT exit, 14-8
$RMEXSTR exit, 14-3, 14-11
$RMPB01S exit NCL procedure, C-4
    consideration, C-5
    example, C-5
    return codes, C-5
$RMPB06S exit NCL procedure, C-2
    consideration, C-2
    example, C-3
    return codes, C-2
$SERVICE system image, 2-7
$TEMPLAT system image, 9-1
    merge, 9-3
    setting up default version, 9-3
$ZREV, 16-12

A

A (Activate) command, 13-23
ACB (access method control block), automating control of, 5-9
access method control block. *See* ACB
ACKLNKFAIL command, 13-26
actions, message rules, 17-11
activation
    consolidated console message profiles, 20-19
    preprocessing, 5-28, 6-10
    process return codes, 5-13
    service, 3-22
Activation Details panel, 5-12
activity log, 3-13
    accessing, 13-28
    accessing from transient log, 13-29
    browsing aids, 13-28
actual states, 3-6
    overriding, 13-19
    RECOVERED, 5-26
        example, 5-26
    service, 6-8

adding
availability map definitions, 5-11, 7-4
EventView message rules, 16-6
EventView rule definitions, 16-5
icon definitions, 12-7
icon panel definitions, 12-16
message group rule definitions, 16-7
message profile definitions, 20-8
process definitions, 8-6
resource definitions, 5-5
loaded system image, to, 3-15
resource group definitions, 12-3
rule set definitions, 16-3
rule set initial action definitions, 16-9
schedule entry lines, 18-5
service definitions, 6-3
status monitor filter definitions, 11-3
system image definitions, 4-4
templates, 9-5
timer rule definitions, 16-6, 18-2
Advanced Operations Management. *See* AOM
alert history, 22-6
alert monitor, 15-4
accessing, 22-2
format, 22-3
Alert Monitor panel, 22-2
alerts, 15-4, 22-1, 22-4
associated details, 22-4
closure, 22-6
filters, 22-3
definitions, 22-3
generating, 15-10
operator notes, 22-5
problem tickets, 22-5
responding to, 22-4
sorting, 22-3
status monitor, and, 22-5
transient log, and, 22-5
ALL C command, 10-5
ALL D command, 10-5
ALL SI command, 10-2
alternate resource relationship, 5-48
OPSYSIDS parameter group, 5-48
AOM (Advanced Operations Management), messages, 21-7
AOMMSG profile parameter, 21-10
AOMPRFSN profile parameter, 21-6
Apply Template panel, 9-6
ARM (automatic restart manager)
resources registered to, working with, 13-3
ASA (Actual State ACTIVE) command, 13-19
ASD (Actual State DEGRADED) command, 13-19
ASF (Actual State FAILED) command, 13-19
ASI (Actual State INACTIVE) command, 13-19
ASU (Actual State UNKNOWN) command, 13-19
authorization, consolidated console, 21-7
AutoAssist, 3-3
facilities, 5-5
AUTOHOLD command, 21-5
AUTOMATED operation mode, 3-8, 13-2
automatic restart manager. *See* ARM
automation
event-based, 15-3
failed, recovering from, 13-25
mechanisms, 1-3
relationships, 3-4
resources, of, 3-4
startup, shutdown, and recovery, 3-4
top-down approach, 2-2
automation engine, 3-3
Automation Log Details panel, 5-29, 6-10
fields, 5-29
automation model, resource-based approach, 2-4
Automation Services command
parameters, A-2, A-3
specifying as a response to a
message, 17-11
timer, 18-7
specifying as a response to a message
example, 17-11
Automation Services, products, 1-2
availability
service and its members, 3-5, 3-21
statistics, 13-32
availability map definitions
accessing, 7-3
adding, 5-11, 7-4
maintenance, 7-10
Availability Map List panel, actions that can be performed from, 7-4
Availability Map panel, 5-10, 6-5

availability maps, 3-9, 7-2
example, 7-6, 7-9
manual override reset, 7-5
map definition, 7-2
overriding specified requirements, 13-21
processes, scheduling, 7-5
resources
attaching, 7-8
detaching, 7-10
rules, 7-3
services
attaching, 7-8
detaching, 7-10
template system image, in, 9-7
accessing, 9-7
adding, 9-7
maintaining, 9-7
temporary, 7-3, 13-21
timer definitions, 7-2
adding, 7-5
viewing, 7-7
using, 7-2

B

benefits, product, 1-4
Boolean expression, G-1
Boolean operators, F-4, F-8
description, G-1
logical, 17-5, F-8, G-3
text analysis, using for, 17-5
Boolean operators, relational, F-4, G-2
contain (CT), F-4, G-2
equal to (EQ), F-5, G-2
greater than (GT), F-6, G-2
greater than or equal to (GE), F-5, G-2
less than (LT), F-7, G-2
less than or equal to (LE), F-6, G-2
like (LIK), G-2
not equal to (NE), F-7, G-2
browsing learnt messages, 19-3

C

cache, 14-7
catchup
scheduled items, requesting for, 18-3
window, specifying, 18-3
CHAIN macro, E-6
checkpoint restart, STARTSYS command, 14-12
COLS command, 13-7
command cache, 14-7
command input, 21-3
COMMAND macro, E-8
commands
$RMCONS NCL procedure, and, B-2
generic, 3-9
graphical monitor, 13-16
line commands, A-2, B-2
message rule actions, 17-11
parameters, A-2, A-3
primary commands, A-3, B-2
status monitor, 13-16
commands, issuing from system console, B-1
commands, line
A (Activate), 13-23
ASA (Actual State ACTIVE), 13-19
ASD (Actual State DEGRADED), 13-19
ASF (Actual State FAILED), 13-19
ASI (Actual State INACTIVE), 13-19
ASU (Actual State UNKNOWN), 13-19
D (Display), 13-16
DB (Data Base), 13-20
DSA (Desired State ACTIVE), 13-19
DSI (Desired State INACTIVE), 13-19
DSR (Desired State Reset), 13-19
EDR (Extended Display Reset), 13-17
GR (Graph Relationships), 13-17
GRT (Graph Relationship Tree), 13-17
MA (Mode AUTOMATED), 13-18
MAP, 13-21
MI (Mode IGNORED), 13-18
MM (Mode MANUAL), 13-18
MR (Mode Reset), 13-18
MV (MoVe), 13-22
R (Relationship), 13-17
S (Status) command, 13-16
T (Terminate), 13-23
TF (Terminate Force), 13-23
TRI (TRee Inactivate), 13-20
W (WTOR), 13-18

XF (Extended Function), 13-17
Z (Zoom), 13-7, 13-14
commands, specific
ACKLNKFAIL, 13-26
ALL C, 10-5
ALL D, 10-5
ALL SI, 10-2
COLS, 13-7
EXTDISP, 13-7
FILTER, 13-10
FORMAT, 13-7
GLOBAL, 14-2
LOAD, 14-3
NRDRET, 21-6
PANEL, 13-16
PROFILE, 13-11, 13-16, 13-29
PURGE, 21-5
REFILTER, 13-10
REFORMAT, 13-7
SHUTFORCE, 14-10
SHUTSYS, 14-9
SORT, 13-8
STARTSYS, 14-11
ZOOM, 13-14
COMPARE macro, E-11
comparing
Boolean operators, using, F-4
message text to a specified string, 17-5
components stored in knowledge base, 2-7
CONCAT macro, E-14
Confirm Copy Replace panel, 6-14, 8-20
Confirm System Image Merge panel, 6-15, 8-21
console message consolidation
CCONSOLIDATN parameter group, and, 15-7
description, 15-3, 20-1
message profile panels, 20-8
message profiles
activating, 20-19
adding, 20-8
maintaining, 20-19
multisystem environment, 20-2
consolidated console, 20-1
accessing, 21-8
AOMMSG profile parameter, 21-10
authorization requirements, 21-7
CC PND status, 21-8
leaving, 21-12
message profile, and, 15-8
monitor profile, 21-7
RMCCOC07 message, 21-8
setup requirements, 21-7
troubleshooting, 21-8
UNSOL profile parameter, 21-9
contain (CT) operator, F-4
correlation keys
function, 16-8
grouped messages, matching in, 16-8
Message Delivery panel, specifying on, 17-9
Related Message Group panel, specifying on, 16-8
CPU consumption
instantaneous, reducing, 14-7
system image load, effect of, 14-7
Criteria List for Report Searches panel, I-3

D

D (Display) command, 13-16
Database Synchronization panel, responding to, 13-27
DB (Data Base) command, 13-20
Define Extended Display Attribute panel, 5-39
Define Extended Filter Definitions panel, 5-37
DELALERT macro, E-16
deleting
learnt messages, 19-7
rule sets, 16-11
system images, 4-4
timer schedule items, 18-6
descriptor codes, message, 17-10
function, 17-4
specifying, 17-4
desired state, 3-6
default, 3-21
overriding, 13-19
removing override, 13-19

desired state management, 3-4, 3-13
considerations, 5-27
Display and Heartbeat Details panel, 5-22
display attributes, 13-3, 13-12
display method considerations, 5-23
display process return codes, 5-22
displaying alert history, 22-6
DOM notification, detecting, 5-35
DSA (Desired State ACTIVE) command, 13-19
DSI (Desired State INACTIVE) command, 13-19
DSR (Desired State Reset) command, 13-19

E

EDR (Extended Display Reset) command, 13-17
enterprise level, 2-2
equal to (EQ) operator, F-5
event exits, implementing, 5-38
event level, 2-2
EVENT macro, E-17
event simulation results, 10-4
summary, 10-5
event simulator, 10-2
availability, 10-2
event-based automation, 15-3
event-related actions, defining, 5-38
events
defining, 5-36
display attributes, 5-39, 5-40
exits, 5-38
extended display, 5-40
extended filter, 5-36
information for display on icons, 5-41
related actions, 5-38
documenting, 5-41
managing, 15-2
simulated, 10-2
EventView, 15-2
benefits of using, 15-4
description, 15-2
EventView message rule definitions
adding, 16-6
message text analysis, using, F-2
EventView rule definitions, adding, 16-5
EventView rules
actions
commands, 17-11
system image operation mode, and, 16-5
message rules
best fit, 17-6
message group rules, associating with, 16-8
overlaps, dealing with, 17-6
ranking, 17-6
EventView variables
correlation keys, using as, 16-12, 17-9
function, 16-12
indicator prefix, 16-12
message group rule records, setting in, 16-7
replacement text, using to provide, 16-12
viewing all, 16-12
EVVARGET macro, E-19
EVVARSET macro, E-20
examples
printer display process, 8-9
RECOVERED state, using, 5-26
exclusion fields, message profile definitions, 20-7
EXECNCL macro, E-21
Execute ACKLNKFAIL Command panel, 13-26
execution conditions
EventView message rule, specifying, 17-6
exits
$RMPB01S NCL procedure, C-4
$RMPB06S NCL procedure, C-2
OIM Problem Management application, C-4
SOLVE:Problem application, C-2
supplied, C-1
system image load, 14-3
system image restart, 14-11
system image shutdown, 14-8
EXTDISP command, 13-7
extended display
##RESET##, 5-40
attributes, defining, 5-39
color and icon, 5-40, E-43
icon, and, 12-11
specifying, 5-40
extended filters, defining, 5-36

Extended Function Exit panel, 5-31, 6-10
extended functions, 13-17
EXTRACT macro, E-22

F

features, product, 1-3
AutoAssist, 3-3
automation engine, 3-3
availability map, 3-9
desired state management, 3-4
event-based automation, 15-3
EventView, 15-2
generic commands, 3-9
Management Services, 3-13
monitors, 3-9
multisystem support, 3-11
process definition, 3-9
ResourceView, 3-3
service management, 3-2
ServiceView, 3-5
startup, shutdown, and recovery automation, 3-4
FILTER command, 13-10
filter keyword
status monitor, specifying, 5-41
filters
alerts, 22-3
extended, defining, 5-36
message
EventView message rules, 17-2
resource definitions, 5-36
resource group, 12-2
service, 6-6
status monitor, 11-1
flags, setting the message delivery flag, 17-7
focal point regions, knowledge base synchronization, 3-12
Force Inactivation Details panel, 5-20
forced inactivation process return codes, 5-20
FORMAT command, 13-7

G

GENALERT macro, E-27
General Description panel, 5-7
generating rules for learnt messages, 19-5
GETSTAT macro, E-30
GLOBAL command, 14-2
syntax, 14-2
global operation mode, 3-8, 14-2
operation, effect on, 3-8, 14-2
Global Process Image Definition panel, 8-17
GOTO macro, E-32
GR (Graph Relationships) command, 13-17
graphical monitor, 13-12
accessing, 13-13
availability map
browsing, 13-21
updating, 13-21
commands, 13-16
customizing, 12-2
description, 3-9, 12-1
display attributes, 13-12
display, changing, 13-16
icon panels, 3-23, 13-12
line commands, 13-15
moving the control of a resource, 13-22
resource relationships, information about, 13-17
service or resource definition
browsing, 13-20
updating, 13-20
starting service and resource manually, 13-23
status information about resource or service, 13-16
stopping a parent and its children, 13-20
stopping service and resource manually, 13-23
transient log, accessing, 13-29
WTOR messages, replying to, 13-18
zooming, 13-14
icon panels, and, 13-15
status monitor, and, 13-15
Graphical Monitor panel, 13-13
graphical monitor, overriding functions
actual state override, 13-19
desired state override, 13-19
removing, 13-19
operation mode override, 13-18
removing, 13-18
resource states override, 13-19
greater than (GT) operator, F-6
greater than or equal to (GE) operator, F-5

groups, message, 16-7
GRT (Graph Relationship Tree)
command, 13-17

H

help facilities for messages, 13-31

I

icon color and extended display, 5-40, E-43
icon definitions
accessing, 12-6
adding, 12-7
example, 12-13
icon, editing, 12-9
maintenance, 12-14
multisystem environment, in, 12-2
Icon Editor panel, 12-8
codes for editing icon, 12-9
Icon List panel, 12-6
Icon Panel Definition List panel, 12-15
icon panel definitions
accessing, 12-15
adding, 12-16
attaching resource group to icon on panel, 12-19
changing resource group attachment to icon, 12-19
generated icon panels, and, 12-20
icon
adding, 12-18
deleting, 12-21
editing, 12-20
finding out about, 12-20
moving, 12-19
icon, duplicating, 12-19
maintenance, 12-21
multisystem environment, in, 12-2
icon panel description files in multisystem environment, 12-2
Icon Panel Generator panel, 12-17
icon panels
description, 12-1, 12-15
generating, 12-15
graphical monitor, and, 13-12
multisystem environment, in, 12-2
panel description file, editing, 12-22
icons, 13-14
change resource group attachment, 12-19
description, 12-1, 12-6
editing codes, 12-9
panels, and, 12-6
resource groups, and, 12-6
structure, 13-14
IGNORED operation mode, 3-8
implementation
automation, 3-16
monitoring, 3-15
tasks, typical operations environment, 3-14
Inactivation Details panel, 5-17
inactivation process return codes, 5-18
inactivation, service, 3-22
including rule sets, 16-4, 16-11
inclusion fields, message profile definitions, 20-7
initial actions
execution considerations, 16-10
specifying, 16-9
Initialization Status panel, 14-6
responding to, 13-27
Inter-System Routing. *See* ISR
ISR (Inter-System Routing)
link manager, 20-2
links, 20-2

J

Job Name Specification panel, 20-10, 20-14

K

knowledge base, 3-11
definition classes, H-1
definitions in, 2-7
dynamic update, 13-2
multisystem considerations, 3-14
synchronization, 3-12
focal point regions, 3-12
subordinates, 3-12

L

learning system messages, 19-2
learnt messages
  browsing, 19-3
  generating a rule for, 19-5
  new indicators, resetting, 19-7
  updating, 19-3
less than (LT) operator, F-7
less than or equal to (LE) operator, F-6
line commands, A-2, B-2
  graphical monitor, 13-15
  issuing from system console, B-2
  parameters, A-2
  using $RMCONS NCL procedure to issue, B-3
LOAD command, 14-3
  checkpoint restart, 14-7
  cold or warm load, 14-7
  exit, 14-3
LOAD Command Parameter Specification panel, 14-4
load exit, 14-3
LOCATE command, I-11
logging a message, 17-7
logical resources, 5-44
  example, 5-45
  timeout on response, 5-32
logical state, 3-6, 3-7
logs
  activity, 13-28
    message delivery, 17-7
  alert history, 22-6
  logging options, 13-31
  system, message delivery, 17-7
  transient, 13-28
    merged, 13-31
  types, 13-28

M

MA (Mode AUTOMATED) command, 13-18
macros, 3-9, E-1
  finding references to, I-2
  process definition, for, 8-6
  return code, E-4
  searching for, I-4
  variable as parameter, 8-9
macros, specific
  CHAIN, E-6
  COMMAND, E-8
  COMPARE, E-11
  CONCAT, E-14
  DELALERT, E-16
  EVENT, E-17
  EVVARGET, E-19
  EVVARSET, E-20
  EXECNCL, E-21
  EXTRACT, E-22
  GENALERT, E-27
  GETSTAT, E-30
  GOTO, E-32
  PARSE, E-33
  REPLY, E-36
  RUNPRCSS, E-40
  SETRC, E-41
  SETSTATE, E-42
  SETVARS, E-45
  SMFWRITE, E-46
  SNMPTRAP, E-47
  STARTNCL, E-49
  STOP, E-50
  SUBJOB, E-51
  SUBPRCSS, E-54
  SVAPI, E-56
  SVCMD, E-59
  SYSCMD, E-60
  TRANS, E-63
  WAIT, E-65
  WAITEVNT, E-66
  WAITSTAT, E-69
  WTO, E-72
  WTOR, E-73
management
  resources, of, 1-3, 3-2
Management Services, 3-13
Management Services command
  specifying as a response to a
    message, 17-11
    timer, 18-7
MANUAL operation mode, 3-8, 13-2
manual overrides, reset by availability map, 7-5
MAP command, 13-21
merging system images, 4-6

message delivery
controlling by means of
route code, 17-10
thresholds, 17-8
setting the deliver flag, 17-8
suppression, 17-12
message delivery thresholds
example, 17-9
keeping different counts for different correlation keys, 17-9
setting, 17-8
message descriptor codes, K-1
message group rules, adding, 16-7
message IDs
about, 17-4
specifying
EventView message rule definitions, 17-4
message profile definitions, 20-10
Message Job Specification panel, 20-11, 20-15
job classes, 20-11
job types, 20-11
message levels, 20-11
message types, 20-11
message learning
controlling, 19-2
description, 19-2
learnt messages
browsing, 19-3
deleting, 19-7
updating, 19-3
message rules for learnt messages, generating, 19-5
multiple messages with the same message ID, learning, 19-2, 19-3
new message indicators, resetting, 19-7
message levels, L-1
message monitor
access authorization, 21-2
accessing, 21-8
command input area, 21-3
command line, 21-3
message display flow, 21-5
AUTOHOLD, 21-5
freezing the display, 21-5
NRD cursor deletion, 21-6
NRD messages
deleting, 21-5
handling, 21-5
removing, 21-6
restoring, 21-6
operator profile, 21-2
profiling for consolidated console, 21-7
roll delete area, 21-3
the title line, 21-4
window, 21-3
message prefixes
$AA-, 5-35
$DN-, 5-35
$MN-, 5-35
message presentation, modifying, 17-9
message profile definitions
accessing, 20-8
adding, 20-8
examples, 20-12, 20-15, 20-17, 20-18
message profiles, 15-7, 20-1
activating, 20-19
consolidated console, and, 15-8
customizing for specific jobs, 20-15
customizing for specific messages, 20-12
excluding information, 20-7
implementing, 20-3
including information, 20-7
maintenance, 20-19
name and ID, 20-4
Profile Status field, updating, 20-18
ranges, using, 20-6
resetting wild cards to literal sense, 20-5
rules for defining, 20-4
size considerations, 20-19
status, 20-9, 20-15, 20-19
wild cards, using, 20-5
message routing codes, J-1

message rules, 15-3
See also EventView rules
actions, 17-11
extended message definition, considerations, 5-25
message text analysis, using, F-2
overlap handling, 5-24, 17-6
resource definitions, in, 5-36
searching for, I-6
wild cards in message text, 17-3
Message Specification panel, 20-10, 20-14
message suppression, sample rule sets, 15-3
message text
matching, 17-2
modifying, 17-9
message text analysis, F-1
criteria, specifying, 17-5
examples, F-4
message-based actions
EventView message rules, 17-11
resource definitions, 5-38
messages
color coding in transient log, 13-30
distributing to consolidated consoles, 20-2
extended definition for resources, 5-36
filtering, 1-1
help for, 13-31
local system, on, 5-33
resource definitions, in, 5-33
resource-specific, 5-33
responding to, 5-24
severity, 13-30
system name prefix, 21-6
WTOR
reply from consolidated console, 21-11
reply from status or graphical monitor, 13-18
messages and codes database, 3-13
messages, processing by EventView rules
delivery criteria, specifying, 17-7
enhancing, 15-3
grouping, 16-7
modifying, 17-9
rerouting, 17-10
responding to, 16-7, 17-11
suppression, 17-7, 17-8, 17-12
system console presentation, determining, 17-10
MI (Mode IGNORED) command, 13-18
MM (Mode MANUAL) command, 13-18
Monitor panel, 13-4
monitors, 13-2
alert, 15-4
customizing, 3-23, 11-1
display attributes, 13-3, 13-12
graphical, 3-9, 12-1, 13-12
message consolidation
disabled, 15-10
enabled, 15-8
status, 3-9, 11-1, 13-3
moving the control of a resource, 13-22
MR (Mode Reset) command, 13-18
multisystem environment, 1-2, 3-11
implementation considerations, 3-14
knowledge bases, 3-12
service automation focal point, 3-5
service management, 3-12
MV (MoVe) command, 13-22

N

NCL (Network Control Language), 3-13
NCL procedure, executing using the $NCL process, 5-42
Network Control Language. *See* NCL
network resource management, 1-3
not equal to (NE) operator, F-7
NRD messages, 21-4, 21-5
cursor deletion, 21-6
delete operator message, 21-4
deleting, 21-5
non-roll delete area, 21-4
removing, 21-6
restoring NRDs, 21-6
roll delete area, 21-4
NRDRET command, 21-6

O

OFF operation mode, 3-8

OIM Problem Management application, exit to, C-4

operation modes, 13-2
- AUTOMATED, 3-8
- global, 3-8, 5-9, 6-4, 14-2
- IGNORED, 3-8
- MANUAL, 3-8
- OFF, 3-8
  - effect on relationships, 5-49
- overriding, 13-18
- removing override, 13-18
- resource definition, 5-9
- service, 6-4
- system images, active, 3-8

operation states
- actual, 3-6
- desired, 3-6
- logical, 3-6

operations environment
- implementation tasks
  - automation, 3-16
  - monitoring, 3-15
  - typical, 3-14
- implementing, 3-13

operations environment model, 2-2
- enterprise level, 2-2
- event level, 2-2
- resource level, 2-2
- service level, 2-2

operations statistics
- availability, 13-34
- messages, 13-33
- resource, 13-32
- resource commands, 13-33
- service, 13-32

operations, service-driven, 3-14

OS Codes Specification panel, 20-10, 20-16

override flags, 13-6

overrides
- desired state, removing, 13-19
- operation mode, removing, 13-18

Owner Details panel, 5-31, 6-10

P

PANEL command, 13-16

Panel List panel, 12-22

panels
- icon, 12-1, 12-15
- message monitor window layout, 21-3

panels, resource definition
- Activation Details, 5-12
- Automation Log Details, 5-29
- Availability Map, 5-10
- Define Extended Display Attribute, 5-39
- Define Extended Filter Definitions, 5-37
- Display and Heartbeat Details, 5-22
- Extended Function Exit, 5-31
- Force Inactivation Details, 5-20
- General Description, 5-7
- Inactivation Details, 5-17
- Owner Details, 5-31
- Restart Control Parameters, 5-16
  - manual activation, effect on, 5-16
  - suppressing restart, 5-16
- State Change Exits, 5-28
- Status Monitor Message Details, 5-24

panels, service definition
- Automation Log Details, 6-10
- Availability Map, 6-5
- Extended Function Exit, 6-10
- Owner Details, 6-10
- Service Filters, 6-6
- Service General Description, 6-4
- State Change Exits, 6-10
- State Thresholds, 6-8

panels, specific
Alert Monitor, 22-2
Apply Template, 9-6
Database Synchronization, 13-27
Execute ACKLNKFAIL Command, 13-26
Graphical Monitor, 13-13
Icon Editor, 12-8
Icon List, 12-6
Icon Panel Definition List, 12-15
Icon Panel Generator, 12-17
Initialization Status, 13-27, 14-6
Job Name Specification, 20-10, 20-14
LOAD Command Parameter Specification, 14-4
Message Job Specification, 20-11, 20-15
Message Specification, 20-10, 20-14
Monitor, 13-4
OS Codes Specification, 20-10, 20-16
Panel List, 12-22
Private Message Profile Control, 21-10
Profile Details, 20-9, 20-12
Resource Group Filter Definition, 12-5
Resource Groups List, 12-3
Simulate Message, 10-3
Simulation Results List, 10-4
Statistics, 13-33
Status Monitor Filter, 11-4
Status Monitor Filter List, 11-2
System Specification, 20-9, 20-13
PARSE macro, E-33
primary commands, A-3, B-2
issuing from system console, B-3
parameters, A-3
using $RMCONS NCL procedure to issue, B-3
primary resource relationship, 5-48
OPSYSIDS parameter group, 5-48
Private Message Profile Control panel, 21-10
problem exits, C-2, C-4
problem ticket, raising for an alert, 22-5
problem troubleshooting
automation, 13-25
message profile activation, 20-19
resource, 13-24
Process Definition panel, 8-19
process definitions
accessing, 8-5
adding, 8-6
chaining processes, E-6
examples, 8-9
macros, 8-6
parameters, setting, 8-8
maintenance, 8-16
process steps, example, 8-6
reserved name, $NCL, 8-6
template system image, in, 9-8
accessing, 9-8
adding, 9-8
maintaining, 9-8
Process List panel, actions that can be performed from, 8-5
processes
$LOG parameter, 8-16
$NCL, 5-42
backing-up, 8-16
changing from global to local, 8-20
description, 3-9, 8-1
EventView variable values, passing, 16-12
executing, 8-2
execution search path, 8-4
merging images, 8-21
optional parameters, 8-9
passing values to macros, 8-9
restoring from backup version, 8-19
return code, 8-6
searching for, I-9
testing, 8-15
interactive, 8-15
types, 3-9, 8-3
updating, 8-18
processes, using
availability map, in, 7-5
chaining, E-6
NCL procedures, executing, 5-42
response to a message, specifying as, 5-38, 17-11
response to a timer, specifying as, 18-7
PROFILE command
graphical monitor, for, 13-16
status monitor, for, 13-11
transient log, for, 13-29
Profile Details panel, 20-9, 20-12
PSM (Print Services Manager), 3-13
PURGE command, 21-5

R

R (Relationships) command, 13-17
recovery
  automation, 13-25
  resource, 13-24
REFILTER command, 13-10
REFORMAT command, 13-7
relationships
  primary and alternate, 5-48
    OPSYSIDS parameter group, 5-48
  resources, between, 5-46
REPLY macro, E-36
reporting
  about, I-2
  locating output, I-11
  output formats, I-2
  search output
    sorting, I-11
    using, I-11
  search, performing, I-3
    macros, for, I-4
    message rules, for, I-6
    processes, for, I-9
Reporting Menu panel, I-3
resource classes, user-defined, 5-43
resource definition classes, H-1
resource definition panels
  Activation Details, 5-12
  Automation Log Details, 5-29
  Availability Map, 5-10
  Display and Heartbeat Details, 5-22
  Extended Functions Exit, 5-31
  Force Inactivation Details, 5-20
  General Description, 5-7
  Inactivation Details, 5-17
  Owner Details, 5-31
  Restart Control Parameters, 5-16
    manual activation, effect on, 5-16
    suppressing restart, 5-16
  State Change Exits, 5-28
  Status Monitor Message Details, 5-24
resource definitions
  ACB, controlling, 5-9
  accessing, 5-3
  adding, 5-5
    example, 5-7
    loaded system image, to, 3-15
  AutoAssist facilities, 5-5
  conversion into template, 9-9
  detaching from availability map, 7-10
  main features, 5-6
  maintenance, 5-51
  operation mode, 5-9
  reasons for, 5-2
  RECOVERED actual state, 5-26
    example, 5-26
  system load balancing, 5-50
  templates, 5-2, 5-9
  using timeout on response, 5-32
resource definitions, message rules
  event actions, defining, 5-38
  event exits, defining, 5-38
  events
    detecting, 5-35
    documenting, 5-41
  extended display attributes, defining, 5-39
  extended message definition, 5-36
  extended message filter
    defining, 5-36
    example, 5-36
  message text analysis, using, F-2
  messages, specifying, 5-33
resource events, detecting, 5-35
resource group definitions
  accessing, 12-3
  adding, 12-3
  filter, 12-2
  filter expression, defining, 12-5
  maintenance, 12-6
  multisystem environment, in, 12-2
Resource Group Filter Definition panel, 12-5
Resource Groups List panel, 12-3
resource groups, defining, 12-2
resource level, 2-2
resource list window, action codes, 5-4
resource management, 3-2

resource relationships, 5-46
child, 5-46
defining, 5-47
deletion of a member, effect of, 5-49
disconnecting, 5-47
information about, 13-17
OFF operation mode, effect of, 5-49
parent, 5-46
primary and alternate, 5-48
OPSYSIDS parameter group, 5-48
resource starting, effect on, 5-48
resource stopping, effect on, 5-48
resource starting
IGNORED operational mode, effect of, 5-48
MANUAL operational mode, effect of, 5-48
resource relationships, effect of, 5-48
sequence of events, 5-12
resource stopping
IGNORED operational mode, effect of, 5-49
MANUAL operational mode, effect of, 5-49
resource relationships, effect of, 5-48
resource templates
<s, using, 9-5
>s, using, 9-5
_, using, 9-5
accessing definitions, 9-4
adding, 9-5
creating and maintaining, 9-4
maintenance, 9-6
updates, applying, 9-6
variable substitution, 9-5
variables
defining, 9-5
using, 9-5
resource templates, availability maps, 9-7
accessing, 9-7
adding, 9-7
maintenance, 9-7
resource templates, processes, 9-8
accessing, 9-8
adding, 9-8
maintenance, 9-8
resources, 5-1
actual state override, 13-19
ARM, registered to, 13-3
availability map, detaching from, 7-10
availability, and service, 3-21
desired state override, 13-19
removing, 13-19
failed, recovering, 13-24
logical, 5-44
moving the control of, 13-22
operation mode override, 13-18
removing, 13-18
operations statistics, 13-32
reasons for defining, 5-2
restart control, 5-14
starting manually, 13-23
status information about, 13-16
stopping manually, 13-23
transient log, 13-29
types, 5-2
ResourceView, 3-3
Restart Control Parameters panel, 5-16
manual activation, effect on, 5-16
suppressing restart, 5-16
restart control, resources, 5-14
restart exit, system image, 14-11
return codes
$RMPB01S exit NCL procedure, C-5
$RMPB06S exit NCL procedure, C-2
activation process, 5-13
display process, 5-22
forced inactivation process, 5-20
inactivation process, 5-18
macro, E-4
processes, of, 8-6
roll delete area, 21-4
route codes, message, 17-10, J-2
function, 17-4
specifying, 17-4
rule set definitions
adding, 16-3
maintenance, 16-11

rule sets
associating timers with, 18-2
associating with a system image, 16-2, 16-5
deleting, 16-11
function, 16-2
implementing, 15-5
including in another rule set, 16-2, 16-11
initial actions, 16-9
maintenance, 16-11
message group rules, 16-7
message rules, 16-6
sample, message suppression, 15-3
status monitoring, 16-4
testing, 16-3
timer rules, 16-6
using, 16-2
rules, message-based actions, 17-11
RUNPRCSS macro, E-40

S

S (Status) command, 13-16
scheduled operations changes
modifying, 13-21
overriding specified changes, 13-21
schedules, timer
details
specifying, 18-4
updating, 18-6
entry lines
adding, 18-5
deleting, 18-6
next execution display, viewing, 18-6
Search for Macro panel, I-4
Search for Message Rule panel, I-6
Search for Process panel, I-9
Search Results panel
macros, I-5
message rules, I-8
processes, I-10
service definition panels
Automation Log Details, 6-10
Availability Map, 6-5
Extended Functions Exit, 6-10
Owner Details, 6-10
Service Filters, 6-6
Service General Description, 6-4
State Change Exits, 6-10
State Thresholds, 6-8
service definitions
accessing, 6-2
adding, 6-3
availability map, detaching from, 7-10
backup service image, restoring from, 6-13
maintenance, 6-11
Service Filters panel, 6-6
Service General Description panel, 6-4, 6-14
Service Image Definition panel, 6-12
Service Image List panel, 6-12
service level, 2-2
Service List panel, actions that can be performed from, 6-2
service-driven operations, 3-14
services, 2-5, 3-21
activation, 3-22
actual states, 6-8
overriding, 13-19
automation focal point system, 3-5
availability, 3-5, 3-21
availability map, detaching from, 7-10
backing-up, 6-11
degraded, 3-22
desired state
overriding, 13-19
removing override, 13-19
failed, 3-22
filters and members, 6-7
inactivation, 3-22
members
fixed weight, 6-7
proportional weight, 6-7
weight criteria, 6-7
merging service images, 6-15
multisystem environment, in, 3-12
operation mode, 6-4
overriding, 13-18
removing override, 13-18
operations statistics, 13-32
pre-activation processing, 6-10
starting manually, 13-23
state thresholds, 6-8
example, 6-9
status determination, 3-22
status information about, 13-16
stopping manually, 13-23
transient log, 13-29
updating from a service image, 6-13

ServiceView, 3-5
SETRC macro, E-41
SETSTATE macro, E-42
SETVARS macro, E-45
shutdown exit, 14-8
SHUTFORCE command, 14-10
  exit, 14-8
  syntax, 14-10
SHUTSYS command, 14-9
  exit, 14-8
  STARTSYS command, and, 14-12
  syntax, 14-9
Simulate Message panel, 10-3
simulated event definitions
  accessing, 10-2
  deleting all, 10-5
simulated events, 10-2
  defining, 10-3
  generating, 10-2
  results, 10-4
    summary, 10-5
Simulation Results List panel, 10-4
size, merged transient log, 13-32
SMFWRITE macro, E-46
SNMPTRAP macro, E-47
SOLVE:Problem application, exit to, C-2
SORT command, 13-8, I-11
  types of sorts, 13-8
Sort Criteria panel, I-11
starting a service or resource manually, 13-23
STARTNCL macro, E-49
STARTSYS command, 14-11
  checkpoint restart, 14-12
  exit, 14-11
  SHUTSYS command, and, 14-12
  syntax, 14-11
startup, shutdown, and recovery automation, 3-4
State Change Exits panel, 5-28, 6-10
state thresholds
  service filters without members, effect of, 6-8
  service, and, 6-8
State Thresholds panel, 6-8
states, overriding, 13-19
statistics
  availability, 13-32
  EventView, gathering for, 16-4
Statistics panel, 13-33
status monitor, 3-9, 11-1, 13-3
  accessing, 13-4
  availability map
    browsing, 13-21
    updating, 13-21
  commands, 13-16
  customizing, 11-2
  display attributes, 13-3
  display, changing, 13-11
  information
    organizing, 13-7
    provided, 13-5
  link failure errors, clearing, 13-26
  override flags, 13-6
  parent and its children, stopping, 13-20
  resource relationships, information about, 13-17
  resource, moving the control of, 13-22
  service or resource
    starting manually, 13-23
    stopping manually, 13-23
  service or resource definition
    browsing, 13-20
    updating, 13-20
  status information about resource or service, 13-16
  stopping service or resource manually, 13-23
  transient log, accessing, 13-29
  variations, 13-3
  views, 11-4
  WTOR messages, replying to, 13-18
status monitor filter definitions
  accessing, 11-2
  adding, 11-3
  example, 11-5
  filter expression, defining, 11-5
  maintenance, 11-6
  multisystem environment, in, 11-2
  refreshing after updates, 13-7, 13-10
Status Monitor Filter List panel, 11-2
Status Monitor Filter panel, 11-4
status monitor filters, 3-23, 11-1, 11-2, 13-10
  filter keyword, specifying, 5-41
  refreshing after updates, 13-7, 13-10
  selecting, 13-10

Status Monitor Message Details panel, 5-24
extended message definition, considerations, 5-25
status monitor, overriding functions
actual state override, 13-19
desired state override, 13-19
removing, 13-19
operation mode override, 13-18
removing, 13-18
resource states override, 13-19
STOP macro, E-50
stopping a service or resource manually, 13-23
SUBJOB macro, E-51
subordinates, knowledge base synchronization, 3-12
SUBPRCSS macro, E-54
SVAPI macro, E-56
SVCMD macro, E-59
SYSCMD macro, E-60
system command
specifying as a response to a message, 5-38, 17-11
timer, 18-7
system console
commands, issuing from, B-1
line command, issuing, B-3
primary command, issuing, B-3
transient log, viewing, B-1, B-5
VTAM failure, and, B-1
system image definitions
accessing, 4-3
adding, 4-4
maintenance, 4-6
reasons for, 4-2
System Image List panel, actions that can be performed from, 4-4
system images
$SERVICE, 2-7
components, 3-20
CPU consumption, 14-7
deleting, 4-4
description, 2-6, 4-1
examples, 4-2
implementing, 3-18
initialization status, 14-6
load exit, 14-3
loading, 14-3
$RMEXSTR exit, 14-3
checkpoint restart, 14-7
cold or warm, 14-7
maintenance, 4-6
merge, 4-6
reasons for, 4-2
restart exit, 14-11
restarting, 14-11
$RMEXSTR exit, 14-11
rule sets, associating, 16-2
shutdown exit, 14-8
shutting down, 14-8
$RMEXSHT exit, 14-8
swapping, 14-6
system images, active
operations modes, 3-8
EventView rule actions, effects on, 16-5
resource definitions, adding, 3-15
variables, D-8
system load balancing, 5-50
system name, 17-4
message prefix, 21-6
system resource management, 1-3
System Specification panel, 20-9, 20-13

T

T (Terminate) command, 13-23
templates
  $TEMPLAT system images
    copying, 9-3
    merging, 9-3
  conversion from resource definition, 9-9
  product specific, 9-2
  system, setting up, 9-3
  types, 9-2
  updates, applying, 9-6
  USRCLS class, 9-2
  versions, 9-3
    default, setting up, 9-3
TF (Terminate Force) command, 13-23
thresholds
  message delivery, 17-8
    example, 17-9
  service states, 6-8
time zones, merged transient log, 13-32
timeout on response, 5-32
  logical resource, 5-32
timer actions
  example, 18-7
  specifying, 18-7
timer rules
  active, listing, 18-8
  adding, 16-6, 18-2
  function, 18-2
  schedule item details, specifying, 18-4
  scheduled actions, 18-7
timer schedule
  entering, 18-4
  function, 18-2
TRANS macro, E-63
transient log profile
  message severity, 13-30
  message source, 13-30
  resetting, 13-30
transient logs, 13-28
  $RMCONS NCL procedure, and, B-5
  accessing, 13-29
  color coding for messages, 13-30
  merged, 13-31
    size, 13-32
    time zones, 13-32
  message help, 13-31
  message severity, 13-30
  printing, 13-31
  resetting, 13-31
  specifying what to display, 13-29
  system console, viewing from, B-1
  user-defined log messages, 13-31
TRI (TRee Inactivate) command, 13-20
trouble tickets, raising for an alert, 22-5

U

UNSOL profile parameter, 21-9
updating
  learnt messages, 19-3
  timer schedule details, 18-6
user-defined extended display
  displaying, 13-7
  removing, 13-17
user-defined log messages, displaying, 13-31
user-defined variable
  icon, and, 12-11
  resource definition, in, 5-41
USRCLS resource class, 5-2
  subclasses, creating, 5-43

V

variables, D-1
  active system image, D-8
  EventView, 1-1
  knowledge base data, D-2
  macro parameter, and, 8-9
  message, D-9
  resource template definitions, used in, 9-5
  status, D-6
views, status monitors, 11-4
VTAM failure, B-1

W

W (WTOR) command, 13-18
WAIT macro, E-65
WAITEVNT macro, E-66
WAITSTAT macro, E-69
wildcard characters
    #, 20-5
    *, 20-5
    @, 20-5
    resetting to literal sense, 20-5
    specifying, 17-4
    using, 17-3
window (screen display), message monitor, 21-3
WTO macro, E-72
WTO, multiline, 17-4
WTOR macro, E-73
WTOR messages
    consolidated console, reply from, 21-11
    status or graphical monitor, reply from, 13-18

X

XF (Extended Function) command, 13-17

Z

Z (Zoom) command, 13-7, 13-14