Tivoli® System Automation for z/OS

Version 3 Release 2

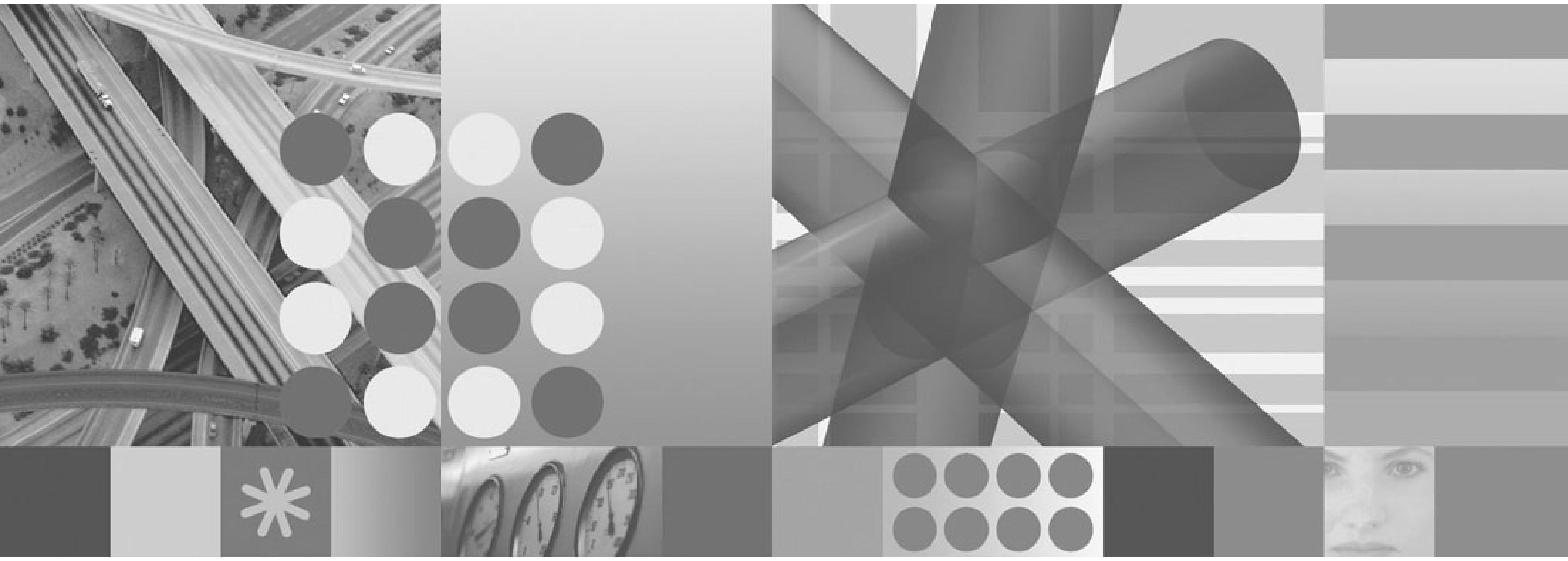
Defining Automation Policy

SC33-8262-05

Tivoli® System Automation for z/OS

IBM

Version 3 Release 2



Defining Automation Policy

SC33-8262-05

Note!

Before using this information and the product it supports, read the information in “Notices” on page xiii.

This edition applies to IBM Tivoli System Automation for z/OS (Program Number 5698-SA3) Version 3 Release 2, an IBM licensed program, and to all subsequent releases and modifications until otherwise indicated in new editions.

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

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

IBM Deutschland Research & Development GmbH
Department 3248
Schoenaicher Strasse 220
71032 Boeblingen
Germany

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

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

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

Contents

Chapter 6. Policy Database Maintenance 317

Chapter 7. Creating Reports about Policy Databases 329

Chapter 8. Building and Distributing Configuration Files 337

Chapter 9. Data Management. 351

Chapter 10. Automating Networks . . 375

Chapter 11. How System Operations Coordinates with Automatic Restart Manager 381

Chapter 12. Conversion Function . . . 383

Appendix A. Concurrent Customization by Multiple Users . . . 387

Appendix B. More about Using the Customization Dialog 391

Appendix C. Example New Policy Objects via File Updates 403

Glossary 405

Index 425

Figures

Tables

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

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

IBM Deutschland Research & Development GmbH
Department 3248
Schoenaicher Strasse 220
71032 Boeblingen
Germany

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

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

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

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

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

Programming Interface Information

This publication documents information that is *not* intended to be used as a programming interface of IBM Tivoli System Automation for z/OS.

Trademarks

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

CICS
CICS Connection
DB2
eServer
GDPS
IBM
IMS
MVS
MVS/ESA
NetView
OS/390
Parallel Sysplex
RACF
RMF
S/390
Sysplex Timer
Tivoli
VTAM
WebSphere
z/OS
zSeries

The following terms are trademarks of other companies:

- LINUX is a registered trademark of Linus Torvalds and others.
- Microsoft is a registered trademark of Microsoft Corporation in the United States and other countries.
- UNIX is a registered trademark of The Open Group in the United States and other countries.

About This Book

This book describes Tivoli® System Automation for z/OS® (SA z/OS). This product plays a key role in supplying high-end automation solutions whereas Tivoli Systems (Tivoli Management Environment, Tivoli) delivers integrated cross-platform management functions. This book contains information about how to enable SA z/OS's automation.

Who Should Use This Book

This book is primarily intended for automation programmers responsible for defining and maintaining the automation policy.

Notes on Terminology

MVS™

References in this book to *MVS* refer either to the MVS/ESA™ product or to the MVS element of z/OS.

NetView

The term *NetView* used in this documentation stands for *IBM® Tivoli NetView for OS/390®*.

OPC

References in this book to *OPC* refer to Tivoli Workload Scheduler (TWS).

Resources, Applications, and Subsystems

These three terms are used in the SA z/OS library to designate automated entities. The correct term to be used depends on the context. See the chapter "Introduction" in *IBM Tivoli System Automation for z/OS Operator's Commands* for the definition of these terms.

Where to Find More Information

The System Automation for z/OS Library

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

Table 1. System Automation for z/OS Library

Title	Order Number
IBM Tivoli System Automation for z/OS Planning and Installation	SC33-8261
IBM Tivoli System Automation for z/OS Customizing and Programming	SC33-8260
IBM Tivoli System Automation for z/OS Defining Automation Policy	SC33-8262

Table 1. System Automation for z/OS Library (continued)

Title	Order Number
IBM Tivoli System Automation for z/OS User's Guide	SC33-8263
IBM Tivoli System Automation for z/OS Messages and Codes	SC33-8264
IBM Tivoli System Automation for z/OS Operator's Commands	SC33-8265
IBM Tivoli System Automation for z/OS Programmer's Reference	SC33-8266
IBM Tivoli System Automation for z/OS CICS Automation Programmer's Reference and Operator's Guide	SC33-8267
IBM Tivoli System Automation for z/OS IMS Automation Programmer's Reference and Operator's Guide	SC33-8268
IBM Tivoli System Automation for z/OS TWS Automation Programmer's Reference and Operator's Guide	SC23-8269
IBM Tivoli System Automation for z/OS End-to-End Automation Adapter	SC33-8271
IBM Tivoli System Automation for z/OS Monitoring Agent Configuration and User's Guide	SC33-8337

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

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

SA z/OS Home Page

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

Related Product Information

You can find books in related product libraries that may be useful for support of the SA z/OS base program by visiting the z/OS Internet Library at http://www.ibm.com/servers/eserver/zseries/zos/bkserv/

Using LookAt to look up message explanations

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

You can use LookAt from these locations to find IBM message explanations for z/OS elements and features, z/VM®, VSE/ESA™, and Clusters for AIX® and Linux™:

- The Internet. You can access IBM message explanations directly from the LookAt Web site at http://www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/.
- Your z/OS TSO/E host system. You can install code on your z/OS or z/OS.e systems to access IBM message explanations using LookAt from a TSO/E command line (for example: TSO/E prompt, ISPF, or z/OS UNIX® System Services).
- Your Microsoft® Windows® workstation. You can install LookAt directly from the *z/OS Collection* (SK3T-4269) or the *z/OS and Software Products DVD Collection* (SK3T4271) and use it from the resulting Windows graphical user interface

(GUI). The command prompt (also known as the DOS > command line) version can still be used from the directory in which you install the Windows version of LookAt.

- Your wireless handheld device. You can use the LookAt Mobile Edition from http://www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/lookatm.html with a handheld device that has wireless access and an Internet browser (for example: Internet Explorer for Pocket PCs, Blazer or Eudora for Palm OS, or Opera for Linux handheld devices).

You can obtain code to install LookAt on your host system or Microsoft Windows workstation from:

- A CD-ROM in the *z/OS Collection* (SK3T-4269).
- The *z/OS and Software Products DVD Collection* (SK3T4271).
- The LookAt Web site (click **Download** and then select the platform, release, collection, and location that suit your needs). More information is available in the LOOKAT.ME files available during the download process.

Summary of Changes for SC33-8262-05

This document contains information previously presented in System Automation for z/OS V3R2.0 Defining Automation Policy, SC33-8262-04.

New Information

APPLICATION SYMBOLS policy item

The APPLICATION SYMBOLS policy item allows you to enter or update application symbols that can be set to various application-specific values. See “APPLICATION SYMBOLS Policy Item” on page 177 for more details.

Changed Information

Updated policy items

The following policy items have been changed:

1. SYMBOL DESCRIPTION for entry type ENT, see “SYMBOL DESCRIPTION Policy Item” on page 97
2. AUTOMATION SYMBOLS for entry type SYS, see “AUTOMATION SYMBOLS Policy Item” on page 127
3. APPLGROUP INFO for entry type APG, see “APPLGROUP INFO Policy Item” on page 139

You may notice changes in the style and structure of some content in this document—for example, headings that use uppercase for the first letter of initial words only, and procedures that have a different look and format. The changes are ongoing improvements to the consistency and retrievability of information in our documents.

This document contains terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Chapter 1. Overview

Before you can use SA z/OS to automate, monitor, and control the resources in your enterprise, you must define your enterprise's automation policy. For this purpose you use the SA z/OS customization dialog to provide information to SA z/OS such as:

- Which resources (such as systems or applications) you want to automate, monitor and control
- How resources are to be associated (grouped) with each other for automation
- The dependencies between resources and groups of resources
- How and what automation, such as automatic startup or shutdown, is to be applied to these resources
- Conditions that automated actions should occur under, and what actions should be performed
- When automation is active, and how it handles certain tasks and events during system operation

The customization dialog leads the SA z/OS policy administrator through the process of defining resources and automation for the enterprise. SA z/OS automatically extracts the required information from this to display resources graphically on the NMC workstation.

Before you begin to use the customization dialog to define your enterprise automation policy, you need to determine how you want SA z/OS to work within your enterprise. Making this determination is part of the process of planning to use SA z/OS. There are many considerations for planning to use SA z/OS that should be aware of. For this purpose, also refer to the planning instructions given in *IBM Tivoli System Automation for z/OS Planning and Installation*.

Note:
Do not use the customization dialogs on a Kanji terminal as they will not work. If you have your terminal defined as a 3278KN terminal, you can circumvent the Kanji restriction by going into ISPF option 0.1 (panel ID ISPOPT1) and redefining the TERMINAL TYPE to 3278.

The remainder of this chapter deals with the following topics:

- "How to Define Automation Policy: An Outline" on page 2 gives an outline of the process of creating enterprise automation policy.
- "The Customization Dialog Primary Menu" on page 2 introduces the Customization Dialog Primary Menu.
- "Entry Type Introduction" on page 3 introduces all entry types.
- "Considerations for the SA z/OS Graphical Interface" on page 8 provides some information on how to customize the monitoring aspect of SA z/OS.
- "Considerations for Automation" on page 9 provides some information on how to customize the automation aspect of SA z/OS.
- "Building and Distributing SA z/OS Automation Policy" on page 10 introduces what you need to do once you have entered your policy information (both for monitoring and automation) into the policy database using the customization dialog.

How to Define Automation Policy: An Outline

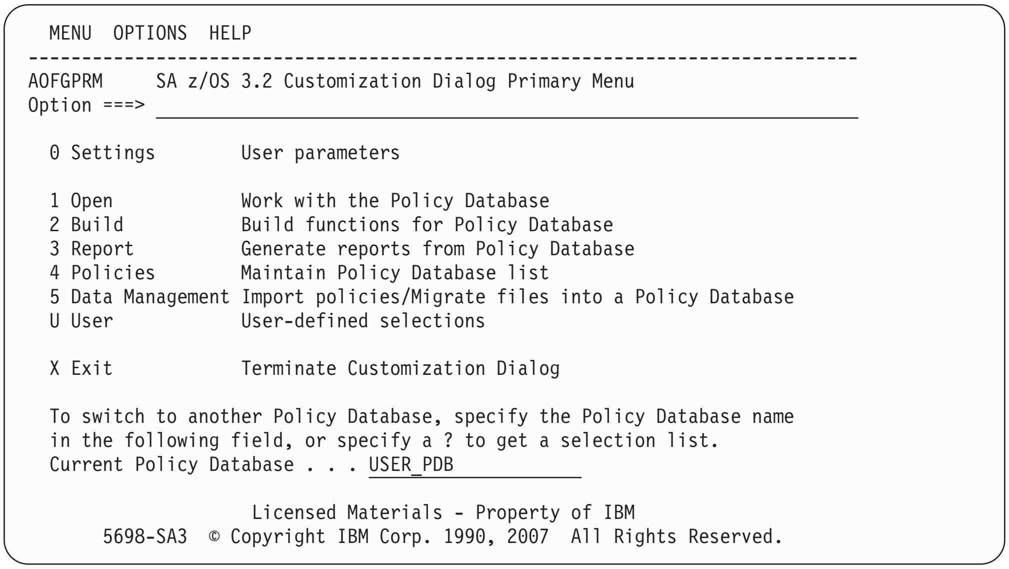
The process of defining automation policy comprises the following steps:

1. At the configuration focal point, where you have the customization dialog installed, create the information in the policy database using the ISPF input panels of the customization dialog.
2. Then use the BUILD function of the customization dialog to produce the required output, the system operations configuration files: Automation Control File (ACF), Automation Manager Configuration File (AMC), NetView Automation Table (AT), and the MPFLSTSA file.
3. Then you distribute the generated output to the sysplexes or standalone systems that you want to automate.

Refer to Chapter 8, "Building and Distributing Configuration Files," on page 337 and Figure 259 on page 346 for more information on building and distributing the generated output files.

The Customization Dialog Primary Menu

Once you have started the customization dialog, you are presented with the Customization Dialog Primary Menu, as shown in Figure 1.



```
  MENU  OPTIONS  HELP
------------------------------------------------------------------------------
AOFGPRM     SA z/OS 3.2 Customization Dialog Primary Menu
Option ===> __________________________________________________________________

  0 Settings        User parameters

  1 Open            Work with the Policy Database
  2 Build           Build functions for Policy Database
  3 Report          Generate reports from Policy Database
  4 Policies        Maintain Policy Database list
  5 Data Management Import policies/Migrate files into a Policy Database
  U User            User-defined selections

  X Exit            Terminate Customization Dialog

  To switch to another Policy Database, specify the Policy Database name
  in the following field, or specify a ? to get a selection list.
  Current Policy Database . . . USER_PDB____________

                   Licensed Materials - Property of IBM
       5698-SA3  © Copyright IBM Corp. 1990, 2007  All Rights Reserved.
```

Figure 1. Customization Dialog Primary Menu

You can select the following options:

0 Settings
This option displays the Customization Dialog Settings Menu, see "Changing Your Customization Dialog Environment" on page 391.

1 Open
With this option, you open the policy database of your choice, see "How to Start the Customization Dialog" on page 34.

2 Build
This option displays the Build Functions Menu, see Chapter 8, "Building and Distributing Configuration Files," on page 337.

3 Report
This option displays the Report Selection Menu, see Chapter 7, “Creating Reports about Policy Databases,” on page 329.

4 Policies
This option displays the Policy Database Selection panel with the list of existing policy databases, see “Creating a New Policy Database” on page 319.

5 Data Management
This option displays the Data Management Menu panel, see “Start Panel for Data Management” on page 351.

U User
This option invokes the User-written Functions Selection Menu. This menu allows you to add installation-specific functions to the customization dialog. For more information refer to “User-written Functions” on page 399.

X Exit Use this option to exit the customization dialog

Entry Type Introduction

All resources in your enterprise, such as systems or applications, or other objects that you use to automate your enterprise data processing, such as components of your z/OS operating system, are called *entry types*. A complete list of SA z/OS entry types is listed on the Entry Type Selection panel (see Figure 2 on page 7), which is the first panel you see after you have decided which enterprise or policy database you are going to work with.

For each entry type you can invoke a series of panels to create, modify, or delete entries of a certain entry type. A single instance of a certain entry type is called a *policy object*. These entries are referred to by their *entry name*.

When you select an object via its name, you get a list of *policy items* specific for the entry type you are working with. For example, the *Timers* entry type may consist of several named sets of timer definitions, each containing a number of timers. These named sets are selected and linked to a particular system as required. Each set can be linked to an arbitrary number of systems and any number of sets can be linked to any particular system.

The numbers and letters in parentheses in the following list indicate the entry type’s index on the Entry Type Selection panel. You can use either numbers or letters to select the appropriate entry type. The alphanumeric index is used in other panels of the customization dialog to identify a certain entry type.

Detailed information and scenarios on how to automate the resources (applications and application groups) and make them available on certain systems are described in Chapter 4, “Scenarios on How to Use the Customization Dialog,” on page 47.

(1/ENT) Enterprise

An Enterprise is the top level of your SA z/OS structure. It comprises enterprise wide data for the complete policy database, including:

- The definition of an operator profile for sending commands
- The definition of INGSEND command parameters
- Processor operations focal point information

(2/GRP) Group

A group is the largest subdivision of an SA z/OS enterprise. An SA z/OS enterprise may have several groups, each consisting of systems or subgroups, or both.

A sysplex group can be used to represent a physical sysplex (systems within the same XCF group ID). A standard group can be used to represent a logical sysplex (systems with different XCF group IDs). The definition of groups is a prerequisite for sysplex-wide automation.

(3/SBG) Subgroup

Like groups, subgroups are collections of systems. You define a subgroup by selecting the member systems and assigning a name to the subgroup. Subgroups can be collected into groups. They offer a finer level of structuring groups.

(4/SYS) System

A system is an image of an operating system running on a host processor monitored by SA z/OS. The system may be MVS, VM, VSE, LINUX, or CF (coupling facility). A system definition includes for example:

- Links to other objects associated with the system, for example, links to application groups
- Application defaults
- System defaults
- Clone specifications

(5/APG) ApplicationGroup

An application group is a set of applications that you assign a name to. You can define two types of application groups: System Application Groups and Sysplex Application Groups. You assign System Application Groups to one or more systems and Sysplex Application Groups to one or more sysplexes.

(6/APL) Application

An application is a z/OS subsystem or job that runs on a system in the SA z/OS enterprise. SA z/OS monitors applications through the messages they issue to the system, MVS commands, and SA z/OS-provided services. An application can belong to more than one application group. An application definition includes:

- Subsystem name and the name of the procedure that runs the job
- Warning and alert thresholds
- Start and Stop specifications
- Individual automation flags

(7/EVT) Events

Events are used for external dependencies where they represent a process that the automation cannot handle. For example, an operator needs to manually edit a file before the application can continue. SA z/OS remembers which events are SET (have occurred) and can be told when to forget them (by defining an UNSET condition).

(8/SVP) Service Periods

Define the availability of applications. A service period is a set of time intervals (service windows) during which a resource (APL, APG, or MTR) should be active or inactive.

(9/TRG) Triggers
In combination with events and service periods, triggers are used to control the starting and stopping of resources (APL, APG, or MTR).

(10/PRO) Processor
A processor is the physical or physically partitioned hardware that a system runs on. To monitor VM guest systems, a ″logical″ processor, called the *ProcOps Service Machine* (PSM), has been introduced.

The OCF-based processors supported by SA z/OS are zSeries® and 390-CMOS processors, excluding the 9221 processor. An OCF-based processor is a processor that uses an operations command facility (OCF) to interact with human operators or external programs to perform operations management functions on the processor. The OCF accepts and processes operations management commands, which SA z/OS sends from the focal point system. The processor operations commands are described in *IBM Tivoli System Automation for z/OS Operator's Commands*.

(11/MTR) Monitor Resource
The Monitor Resource Policy Object enables you to monitor the performance and health of an application. That is, a separate status informs you about the application's health. This health status can be used by the automation manager to make decisions and, if necessary, trigger automation for the subject application. This monitoring can be used to make the currently available JES3 monitoring obsolete.

(20/PRD) Product Automation
With this policy object, you can define automation policy for the following products:

- CICS®
- IMS™
- TWS

(30/TMR) Timers
Automation timer definitions. This object is used to schedule automation tasks which are periodic, or which must run at a particular time.

(31/TMO) Timeout Settings
The timeout values for responses to MVS commands and cross-domain commands. This entry type enables you to create sets of timeout definitions. Each timeout definition can specify timeout values for:

- System, NetView, or SA z/OS commands
- Cross-domain logon requests
- Your own timeout variables

If the time limit is exceeded, it is assumed that an error condition exists and an appropriate action is taken.

(32/TPA) Tape Attendance
Specifies the time during which tape drives are either attended or unattended. This entry type also allows you to specify whether you want to monitor tape mounts on this system. If you do want to monitor tape mounts, you must enter warning and alert thresholds (in minutes and seconds) indicating how long a wait is allowed in mounting a tape before the response is considered unsatisfactory.

(33/MVC) MVS Component
This entry type allows you to apply automation to various z/OS data sets, resources and facilities.

(34/MDF) MVSCOMP Defaults
Sets defaults for z/OS component (entry type MVS Component) automation.

(35/SDF) System Defaults
This entry type allows you to set various defaults that are used throughout the system. The values set here may be overridden by other entry types, such as Application, Application Defaults, or MVSCOMP Defaults, or by setting appropriate values in individual applications or MVS Component objects.

(36/ADF) Application Defaults
This entry type specifies default values for automation flags and error threshold settings. These values are used as the default for any Application object on the system that the Application Defaults set is attached to.

(37/AOP) Automation Operators
This entry type maps automation functions to automation operators. *Automation operators* are automated operator tasks, or *autotasks*, that can respond to operating system, application, and network messages without requiring a human operator. Sets of automation operators can be linked to a system. Each automation operator takes actions similar to those of a human operator, such as monitoring the system and handling and coordinating recovery and shutdown operations.

Automation operators are assigned specific messages to act on.

(38/NFY) Notify Operators
Defines NetView operators who receive notification messages.

(39/NTW) Network
Contains definitions for Gateway, adjacent NetView domains (for earlier releases only), SDF focal points, OMEGAMON sessions, TAF sessions, and authentication.

(40/XDF) Sysplex Defaults
This entry type specifies default values of the desired status and the inform list for sysplex resources.

(41/RES) Resident CLISTs
Automation CLISTs that must be preloaded to improve performance.

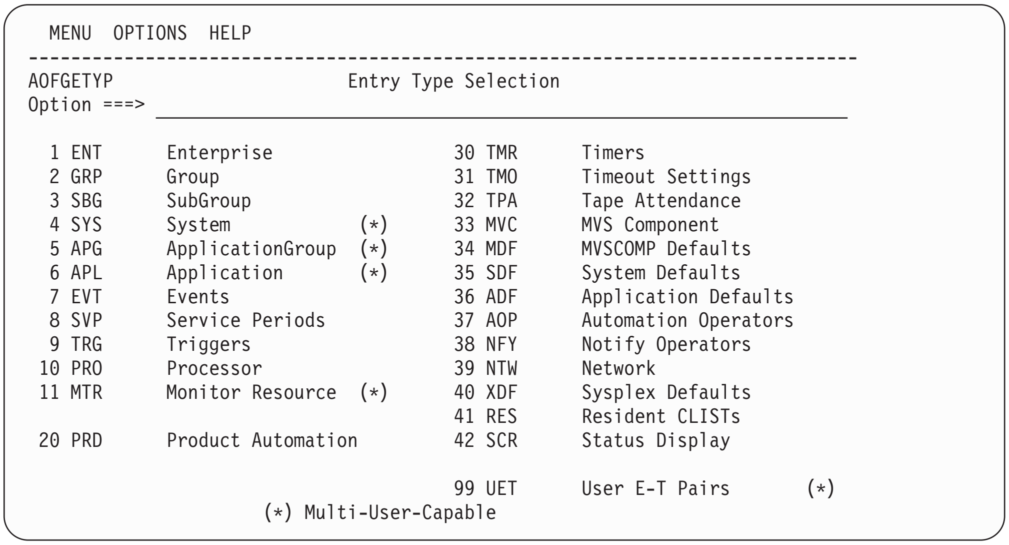
(42/SCR) Status Display
Defines NetView Status Display facility (SDF) settings.

(99/UET) User E-T Pairs
User-defined entry/type data for the automation control file. Used for user-defined extensions to automation policy.

Policy Objects and Policy Items

Each policy object has a number of policy items associated with it. To define policy, you enter data for the policy items of each policy object in the SA z/OS enterprise.

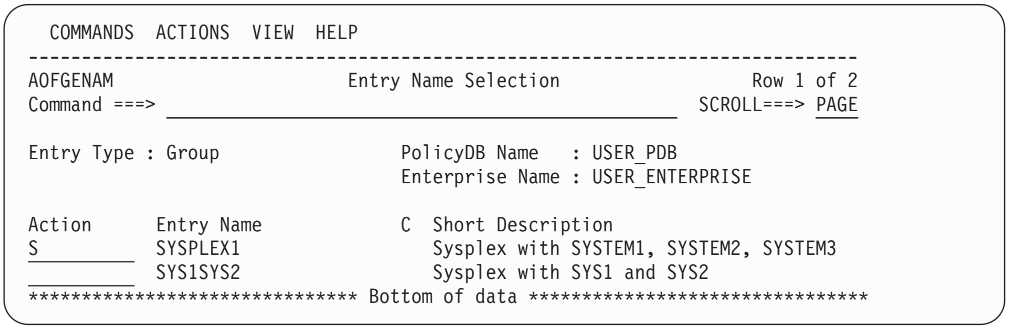
For example, if you choose option 2 on the Entry Type Selection panel (Figure 2 on page 7) to specify Group policy, Figure 3 on page 7 is displayed. (The labels, for example, AOFGETYP, AOFGENAM, are the panel IDs that are displayed after you enter the command `PANELID` on the command line.)



```
  MENU  OPTIONS  HELP
------------------------------------------------------------------------------
AOFGETYP                      Entry Type Selection
Option ===> _________________________________________________________________

  1 ENT      Enterprise                30 TMR      Timers
  2 GRP      Group                     31 TMO      Timeout Settings
  3 SBG      SubGroup                  32 TPA      Tape Attendance
  4 SYS      System             (*)    33 MVC      MVS Component
  5 APG      ApplicationGroup   (*)    34 MDF      MVSCOMP Defaults
  6 APL      Application        (*)    35 SDF      System Defaults
  7 EVT      Events                    36 ADF      Application Defaults
  8 SVP      Service Periods           37 AOP      Automation Operators
  9 TRG      Triggers                  38 NFY      Notify Operators
 10 PRO      Processor                 39 NTW      Network
 11 MTR      Monitor Resource   (*)    40 XDF      Sysplex Defaults
                                       41 RES      Resident CLISTs
 20 PRD      Product Automation        42 SCR      Status Display

                                       99 UET      User E-T Pairs       (*)
                      (*) Multi-User-Capable
```

Figure 2. Entry Type Selection Panel

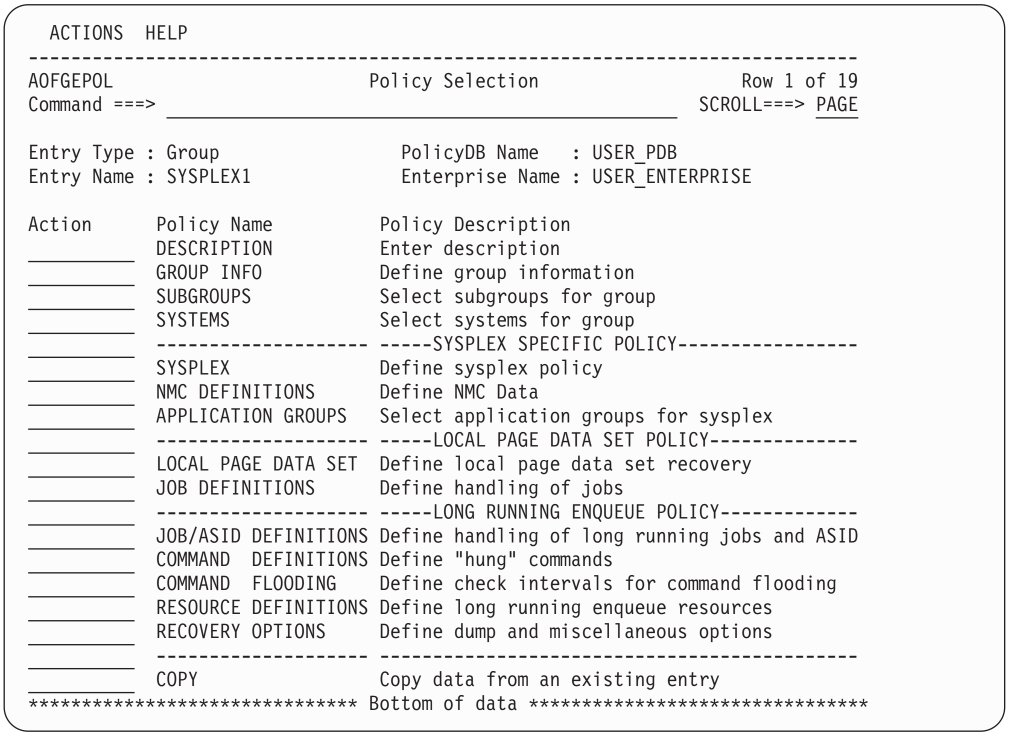
Figure 3 shows the Group policy objects SYSPLEX1 and SYSPLEX2.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGENAM                      Entry Name Selection                Row 1 of 2
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
                                   Enterprise Name : USER_ENTERPRISE

Action      Entry Name             C  Short Description
S_________  SYSPLEX1                  Sysplex with SYSTEM1, SYSTEM2, SYSTEM3
__________  SYS1SYS2                  Sysplex with SYS1 and SYS2
****************************** Bottom of data ********************************
```

Figure 3. Entry Name Selection Panel for Groups

If you select an existing group, the Policy Selection panel is displayed, as shown in Figure 4 on page 8. This panel gives you a list of the policy items for a group. You can enter S for SELECT next to any policy item, to display the appropriate policy input panel. Figure 4 on page 8 shows the Policy Selection panel for groups, listing the policy items for the selected policy object SYSPLEX1.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                  Row 1 of 19
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : SYSPLEX1              Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  GROUP INFO           Define group information
__________  SUBGROUPS            Select subgroups for group
__________  SYSTEMS              Select systems for group
__________  -------------------- -----SYSPLEX SPECIFIC POLICY-----------------
__________  SYSPLEX              Define sysplex policy
__________  NMC DEFINITIONS      Define NMC Data
__________  APPLICATION GROUPS   Select application groups for sysplex
__________  -------------------- -----LOCAL PAGE DATA SET POLICY--------------
__________  LOCAL PAGE DATA SET  Define local page data set recovery
__________  JOB DEFINITIONS      Define handling of jobs
__________  -------------------- -----LONG RUNNING ENQUEUE POLICY-------------
__________  JOB/ASID DEFINITIONS Define handling of long running jobs and ASID
__________  COMMAND  DEFINITIONS Define "hung" commands
__________  COMMAND  FLOODING    Define check intervals for command flooding
__________  RESOURCE DEFINITIONS Define long running enqueue resources
__________  RECOVERY OPTIONS     Define dump and miscellaneous options
__________  -------------------- --------------------------------------------
__________  COPY                 Copy data from an existing entry
****************************** Bottom of data ********************************
```

Figure 4. Policy Selection Panel for Sysplex Groups

Considerations for the SA z/OS Graphical Interface

The customization dialog leads the SA z/OS policy administrator through several panels to define the systems and other automated resources of the enterprise. The resulting *enterprise policy* provides the information that SA z/OS uses to provide objects for the displays on the NMC workstation.

The objects visible on the NMC workstation are based on the resources that are defined for NMC using the inform list specification in the customization dialog and therefore known to SA z/OS. For more details, see “APPLICATION INFO Policy Item” on page 168.

The automation manager component of SA z/OS monitors the resources and their status. SA z/OS permanently puts information about the resources and their status into the focal point RODM from where it is retrieved by the NMC workstation.

Grouping Resources Logically

To provide the hierarchy required for graphical representation on the NMC workstation, you need to define your enterprise in terms of the relationships between the resources. The reason for grouping your enterprise resources within SA z/OS is to simplify the representation of the status information that can be generated by your enterprise. The decisions you make on how to group your resources have a great impact on how powerful the graphic interface on the NMC workstation is for you. Each group (of systems or applications) can be represented by a single icon in the NMC workstation displays. SA z/OS monitoring acts as a filtering mechanism for the status data generated by the components of these groups. The status of the group is determined by the statuses of its members using an algorithm defined independently for each grouping of resources in your SA z/OS enterprise.

The grouping of resources allowed by SA z/OS is flexible. For example, systems may be grouped by geographical region or use or type. You may decide to place all high priority jobs in one application group and specify that the status of the group will be ″degraded″ if only one of its subsidiary applications has an unsatisfactory status.

For information on how to set up the graphical interface on the NMC workstation refer to *IBM Tivoli System Automation for z/OS Planning and Installation*. For information on how to use this graphical interface refer to *IBM Tivoli System Automation for z/OS User's Guide*.

Considerations for Automation

While information for monitoring of resources is defined at the enterprise level, automation of resources and applications is defined at the system level. Each system in an SA z/OS enterprise uses SA z/OS routines to automate operator functions for its own applications and z/OS components. The programs that automate these functions are called *SA z/OS automation procedures*. SA z/OS automation procedures invoke SA z/OS provided *common routines* and *generic routines*. You can customize your automation by writing your own *automation procedures* which can also invoke the SA z/OS provided routines.

SA z/OS provides automation procedures that are internally invoked according to the NetView automation table and that manage, for example, the following:
- WebSphere® Application Server
- SAP
- TCP/IP
- HTTP Server
- Job Entry Subsystem/2 (JES2)
- Job Entry Subsystem/3 (JES3)
- Time Sharing Option/Extensions (TSO/E)
- APPC
- ASCH
- Resource Measurement Facility (RMF™)
- Virtual Telecommunication Access Method (VTAM®)
- NetView
- Transient jobs
- Non-z/OS resources
- Any other applications
- Special automation functions are also provided to manage CICS, IMS, DB2®, and to integrate with TWS

SA z/OS provides automation procedures that enable recovery of the following z/OS components and data sets:
- SYSLOG data sets
- LOGREC data sets
- System Management Facility (SMF) data sets
- Write-to-operator (WTO) buffers
- JES spools
- z/OS dump data sets
- Action Message Retention Facility (AMRF) buffers
- sysplex resources, for example, coupling facility

With SA z/OS you can also define policies for the automated startup and shutdown of processors and for the automated IPL and shutdown of operating systems.

The above lists represent applications and z/OS components which SA z/OS provides automation, but you are not limited to these. You can extend automation by adding new applications, and you can create your own automation procedures.

The SA z/OS policy administrator uses the customization dialog to set the way that enterprise resources are to be handled by SA z/OS when predefined conditions are met. In addition to invoking provided automation procedures, SA z/OS can be used to trigger user-written automation procedures.

Note: In order to avoid problems with automation procedures and customization dialogs, do not use the following terms for entry names or application subsystem names:

RC, RETURN, or SIGL

Building and Distributing SA z/OS Automation Policy

After you have entered all data necessary to define your enterprise monitoring and automation policy in the policy database, you use the BUILD function of the customization dialog to build the files needed by SA z/OS. These include:

- The automation control files for the automation agent (ACF) and automation manager (AMC).
- The processor operations control file and a sample NetView operation definition file.
- NetView automation tables and the MPFLSTSA member.

The automation configuration definitions are needed if you want to use system operations automation functions. If you want to use processor operations automation functions (for example, monitoring processors), the processor operations control file must be built, and the corresponding NetView operator definitions must be provided.

Refer to Chapter 8, “Building and Distributing Configuration Files,” on page 337 for more information on the SA z/OS build function.

The automation configuration files must be distributed to each system in the SA z/OS enterprise where SA z/OS system operations functions are to run. In general, this is all of the z/OS systems in the SA z/OS enterprise. In a sysplex, the data set name of the automation control file must be equal on all systems in the sysplex.

However, it is highly recommended to use a shared data set in a sysplex, and if possible, to use a shared data set for the whole enterprise.

Chapter 2. Automation Concepts

This chapter describes various SA z/OS concepts that you should read before using the customization dialog to define automation policy for your enterprise. The following topics are discussed:

- "Goal Driven Automation"
- "Automatic Calculation of Schedules (Service Periods)" on page 15
- "Relationship Support" on page 15
- "How SA z/OS Uses Error Thresholds" on page 17
- "Automation Flags" on page 20
- "Event and Trigger Support" on page 25
- "Assigning System Automation Symbols (AOCCLONE)" on page 27
- "Nested Classes" on page 29

Goal Driven Automation

SA z/OS implements goal driven automation.

One flavor of goal driven automation in SA z/OS are events and triggers on the one hand and service periods on the other hand (see "Event and Trigger Support" on page 25 and "Automatic Calculation of Schedules (Service Periods)" on page 15). They basically allow you to define goals for applications, application groups, or monitor resources (to be available or unavailable) and to specify external events that, in addition to the defined goals, need to be satisfied before an application, application group, or monitor resource can be really started or stopped.

In SA z/OS you can define the applications, application groups, and monitor resources to be resources (application resources, application group resources and monitor resources) that you want to automate.

Another flavor of goal driven automation is that automation programmers define the default behavior or state of resources using the customization dialog so that SA z/OS tries to keep the resource in the specified state during specified schedules under specified prerequisites.

If operators want to change the goal of a resource, they may issue or remove a request to start or stop it using the INGREQ or INGSET command with appropriate startup or shutdown parameters. A request is executed by SA z/OS only if it does not conflict with requests of higher priority. Otherwise, as requests are persistent, they only take effect when conflicting requests of higher priority are resolved. Operators must remove obsolete requests from SA z/OS.

Note:
INGREQ requests are persistent. They are remembered across session boundaries until they are explicitly revoked.

You can find more details on the concepts of goal driven automation later in this chapter or you can also refer to *IBM Tivoli System Automation for z/OS User's Guide*.

Grouping Support

Modern applications are often composed of many components, such as data servers, networking and security components. Examples are client/server applications where the application logic is distributed between a client and one or more servers and where the data can also be distributed between two and more servers. These components are often spread among the various systems in the sysplex.

In SA z/OS you can automate a complete application group. If you want to have all members (for example, all applications) of an application group available, you just tell SA z/OS to have the application group available.

Figure 5 presents a grouping example. Complex applications that are made of individual application components are denoted with G1–G3 (G stands for group). The application components are denoted with A*n* and B*n*. For example, the complex application group G2 comprises application group G1 and the single applications A4 and A5. Application group G1 in turn consists of three applications A1, A2, and A3. Also, application groups may comprise applications that may or may not be contained in other application groups. G3 is such an example.

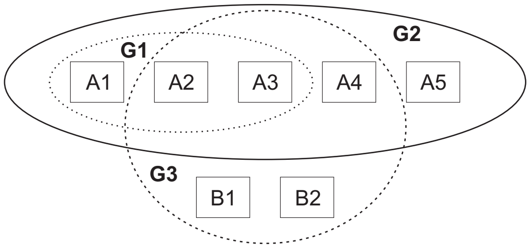


Figure 5. Example of Aggregating Applications into Application Groups

A Grouping Scenario

The following scenario shows how grouping is performed on the conceptual level. For information on how to enter such a grouping policy into the customization dialog, refer to “How to Work with Resources” on page 47.

Grouping applications and nesting application groups is a process that can be logically divided into multiple steps:

- “Step 1: Identify Resources”
- “Step 2: Group the Resources” on page 13
- “Step 3: Nest the Groups” on page 14
- “Step 4: Match Resources to Systems” on page 14

Note:
Applications and application groups are also referred to with the general term *resources*.

Step 1: Identify Resources

Imagine a scenario with two application groups representing a data warehouse and a Web server.

- The data warehouse requires the following resources: VTAM, TCP/IP, DB2 (consisting of components DB2MAIN, DB2IRLM), CICS (consisting of components CICSMAIN, CICSAOR1, CICSAOR2, CICSAOR3, CICSTOR, CICSFOR), and DWH.
- The Web server requires the following resources: VTAM, TCP/IP, DB2 (DB2MAIN, DB2IRLM), and WBSRV.

Step 2: Group the Resources

Some of the needed resources identified in the previous step are candidates for aggregation:

- Resources that are made of multiple components; in our example, these are:
  - CICSAOR1, CICSAOR2 and CICSAOR3 which could be aggregated into a group called CICSAOR, so that they can be automated as a unit.
  - group CICS, which will contain resources CICSMAIN, CICSTOR, CICSFOR and the aggregated CICSAOR group.
  - group DB2, which will contain resources DB2MAIN and DB2IRLM.
- Resources that are used by multiple exploiters; in our example, these are:
  - VTAM and TCP/IP, which are used by the data warehouse and the Web server; so we will aggregate them into a group called NET.
  - DB2MAIN and DB2IRLM, which are used by the data warehouse and the Web server; so we will aggregate them into a group called DB2.
- Finally we conceive the data warehouse as group DWHAPG and the Web server as group WBSRVAPG for later nesting the other groups into them as required (see “Step 3: Nest the Groups” on page 14).

Figure 6 shows a possible aggregation of all components into groups.

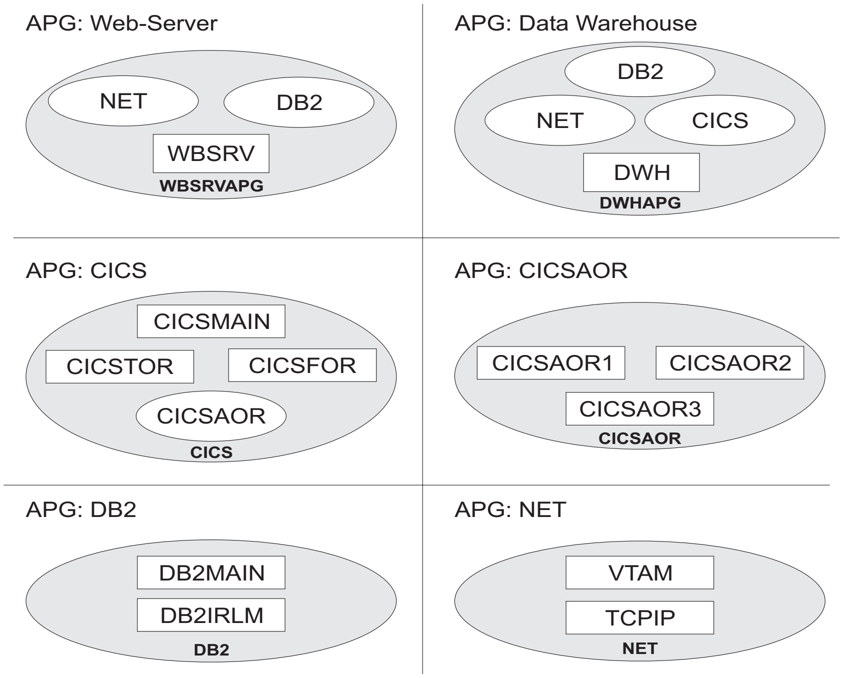


Figure 6. Application Groups

In Figure 6 on page 13, oval labels represent application groups, rectangle labels represent applications. For example, application group CICS contains applications CICSMAIN, CICSTOR, CICSFOR and the nested application group CICSAOR.

Step 3: Nest the Groups

Figure 7 shows how the application groups are nested in this environment. From an operator's point of view, a start request against WBSRVAPG should start all imbedded components in a reasonable order. The same is true for requests against all other groups such as, for example, NET and DB2.

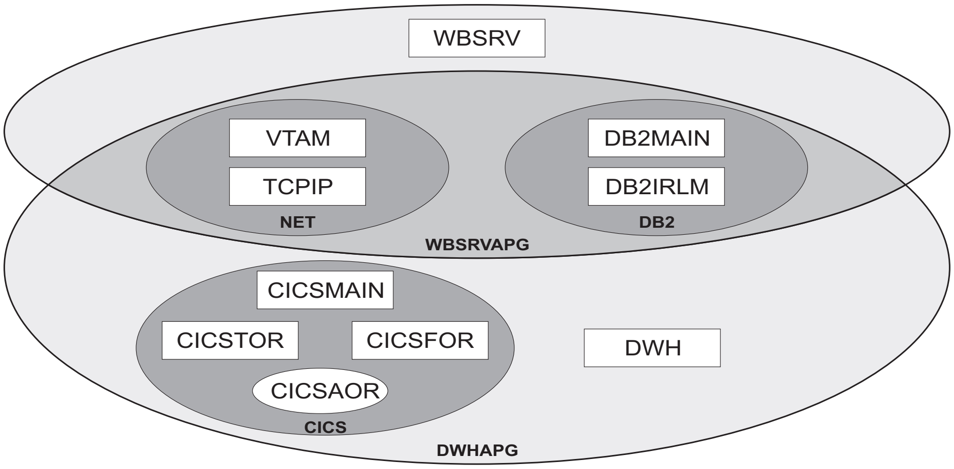


Figure 7. Grouping of Applications into Nested ApplicationGroups

Step 4: Match Resources to Systems

Next, system administrators must decide what systems and sysplexes these groups and applications should run on. Assume that in our scenario we have an environment of three z/OS systems SYSTEM1, SYSTEM2, SYSTEM3 that are members in sysplex SYSPLEX1.

In our example, a possible decision could be:

- The Web server group WBSRVAPG should run on SYSTEM1.
- The data warehouse group DWHAPG should run on SYSTEM2 and (if SYSTEM2 is not available) on SYSTEM3.
- The DB2 group, containing applications DB2MAIN, DB2IRLM should run on SYSTEM2 and on SYSTEM3.
- The NET group, containing applications VTAM, TCPIP should run in the entire sysplex.
- The CICS group, containing applications CICSMAIN, CICSFOR, CICSTOR and the application group CICSAOR should run on SYSTEM2 and on SYSTEM3.

Refer to "How to Work with Resources" on page 47 for detailed information on how to implement this scenario using the customization dialog.

Automatic Calculation of Schedules (Service Periods)

Service periods allow users to schedule the availability of applications, application groups, or monitor resources. A service period is a set of service windows during which a resource (application, application group, or monitor resource) should be active or inactive. So with service periods, you can specify both the up and down times for a resource.

Service periods set the desired status of the applications, application groups and monitor resources that they are linked to.

During the process of evaluating operator requests, SA z/OS calculates schedules for the availability of resources. This means that if you have a resource, say a database, that is supporting two applications, you can configure things so that the database will automatically be made available when either one of the two applications should be available, and to be unavailable otherwise.

If, for example, application A should be up from 7 a.m. to 7 p.m., and application B should be up from 9 a.m. to 3 p.m. and 5 p.m. to 9 p.m., simply setting these schedules for the applications and giving them a start dependency upon the database is sufficient to have SA z/OS calculate a schedule for the database of 7 a.m. until 9 p.m.. Note that this inference is done dynamically at runtime and will also take into account operator requests and schedule overrides.

Adaptive Scheduling

When defining the sysplex policy, the automation programmer can specify that certain resources are backups for other resources. In order to meet or maintain application availability, SA z/OS will activate these resources (if no other requests or policy definitions are opposed). This means that if an application's normal configuration is unavailable, SA z/OS will start an alternate configuration that will provide the same service. This can be configured so that no operator intervention is required.

This feature works best when the automation programmer sets schedules on a group level, rather than on an individual resource level. Applications need to be defined in groups, as it is the selective request propagation by the group resources that achieves the effect.

Relationship Support

One of the most effective automation capabilities of SA z/OS is the concept of *dependency relationships*. You define relationships between *dependent* resources and *supporting* resources.

In the customization dialog you always define relationships from the point of view of the dependent resource, and specify the name of the supporting resource.

Relationships can be specified together with a *condition* that must be satisfied for the *supporting resource*, before the specified action can be processed for the dependent resource.

Examples for relationships are *MAKEAVAILABLE* or *MAKEUNAVAILABLE*. Examples for conditions are *WhenAvailable* or *WhenDown*.

A typical dependency relationship that you might define for a dependent resource, **A**, would be: *MAKEAVAILABLE/WhenAvailable*, where the condition *WhenAvailable* applies to a supporting resource **B** (see Figure 8).

So the relationship defined in Figure 8 would read: Only if the desired state for the dependent resource **A** is available, make it available as soon as the supporting resource **B** is available.

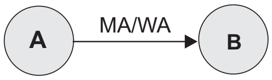


Figure 8. Relationships and Conditions

In the example shown in Figure 8, a request to become available is issued for resource **A**. Resource **A** cannot be made available unless **B** is available. SA z/OS implements the concept of *request propagation*. A request will be propagated along the dependency graph so that ultimately the original request can be fulfilled. In this example, the request to **A** will be propagated to **B**, which can make itself available because it is not dependent on anything else. Then, after **B** is available, **A** will become available. Thus the original request is persistent until it is withdrawn.

Consider another example: assume that **A** has its relationship with the supporting resource **B** defined as HASPASSIVEPARENT, and a request to become available is issued for **A**. This request will not be propagated along the dependency graph. A separate request must be issued for **B** to make the resource **A** available.

There is a special relationship, *HASPARENT*, that simplifies relationship specifications. It is equivalent to a *MAKEAVAILABLE/WhenAvailable* dependency between the dependent and supporting resource and a MAKEUNAVAILABLE/ WhenDown dependency from the supporting resource to the dependent resource, as shown in Figure 9.

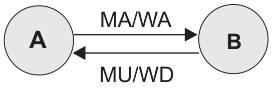


Figure 9. The HASPARENT Relationship

Relationships may be defined between any two resources within the same sysplex. The resources may be on different systems or may be Sysplex Application Groups. Figure 10 on page 17 presents an example of relationships across system boundaries.

Note:

An application group can also be either a supporting resource or a dependent resource. It can even be simultaneously a supporting resource and a dependent resource, as shown in Figure 10 on page 17.

Be careful not to define loops.

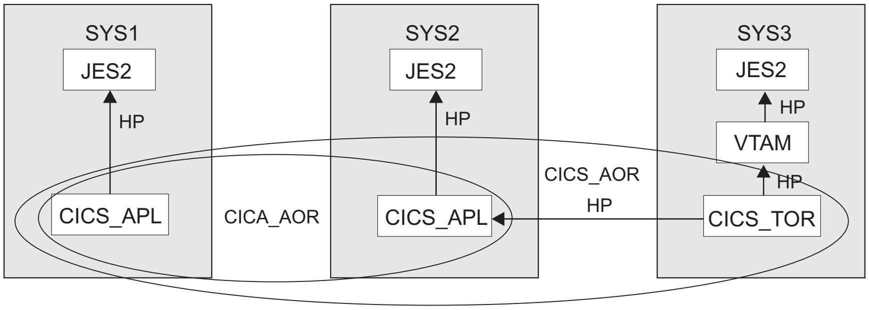


Figure 10. Relationship in a Sysplex

Figure 10 shows an example of grouping and relationships defined across a sysplex consisting of the systems SYS1, SYS2 and SYS3.

In this figure, the arrows denote *HASPARENT* relationships. For example, CICS_TOR will be started as soon as one member of application group CICS_AOR/APG is available and VTAM is available.

If a group is a dependent resource, all *MAKEAVAILABLE* relationships apply to the members of the group as well as to the group as a whole. For example, in Figure 11, resource APL will not start before the resources CICS, VTAM, and JES2 are available.

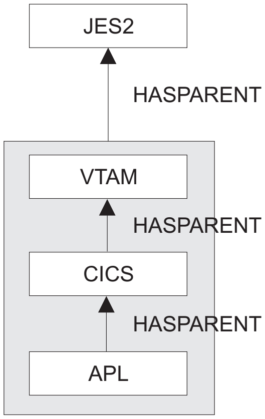


Figure 11. Relationship between Applications and Application Groups

For information on how to define relationships refer to “RELATIONSHIPS Policy Item” on page 143. During automation in your enterprise, conflicting relationships and requests could possibly occur. *IBM Tivoli System Automation for z/OS User’s Guide* describes such a scenario and informs the operator how to handle such a situation.

How SA z/OS Uses Error Thresholds

Error thresholds influence whether SA z/OS recovers from an error situation or not. These are a number of error conditions within a certain time interval, such as five error conditions requiring restart within one hour. During a condition requiring restart, SA z/OS checks whether the number of occurrences of the condition reaches the critical threshold. If it is reached, SA z/OS does not attempt restarting a resource. For applications, you can define a critical threshold for restarting these applications.

For z/OS components, such as dump data sets or log data sets, you can define thresholds to limit the frequency of how often they may be deleted after they ran full without an action being taken or a notification being sent to the operator.

Error thresholds also determine when you should be alerted to problems. The primary use of error thresholds is to track subsystem abends and ensure that the abend and restart cycle does not become an infinite loop, but they may also be customized for other uses.

The following sections inform you how to define error thresholds. Information on how to display threshold settings and occurrences, and to interactively set thresholds (by using the command INGTHRES) is provided in *IBM Tivoli System Automation for z/OS User's Guide*.

Setting Up a Threshold

SA z/OS uses error thresholds to determine when to stop recovering from an error situation. You can define error thresholds to track errors for MVS components and applications. You can set error thresholds for different policy objects using either the THRESHOLDS or MINOR RESOURCE THRES policy item (see Figure 12 on page 19):

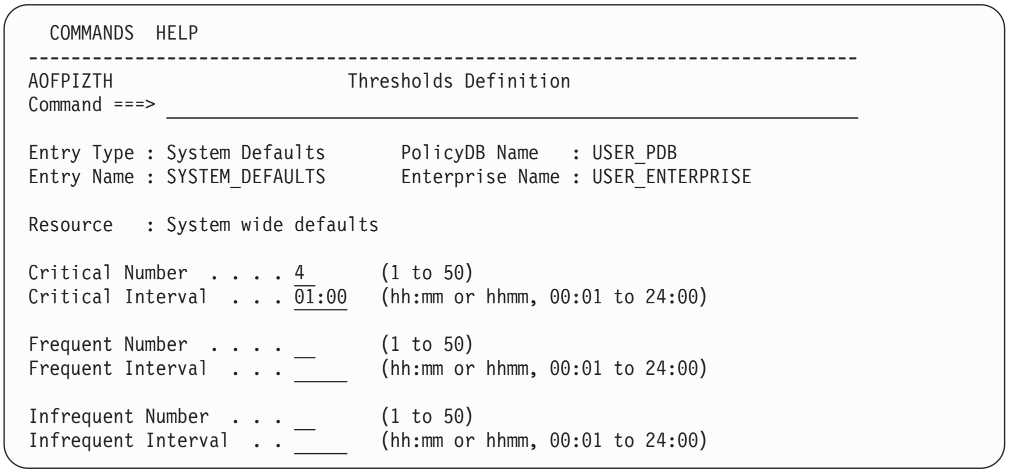
Thresholds can be displayed and modified using the INGTHRES command (see *IBM Tivoli System Automation for z/OS Operator's Commands*).

To set error thresholds for all MVSCOMP resources, choose MVSCOMP Defaults from the Entry Type Selection panel. To set error thresholds for all applications, choose Application Defaults from the Entry Type Selection panel. If you do not specify error threshold defaults for MVS components or applications, SA z/OS uses the error thresholds for all monitored resources, which are defined in the System Defaults objects.

Thresholds are defined by specifying the number of errors in a particular time interval, for example three times in one hour. Figure 12 on page 19 shows the Thresholds Specification panel for the System Defaults object.

Note:

Setting a *critical threshold* to 1 will disable the recovery process already with the first occurrence of the error. This is valid independently from the specified time interval.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIZTH                      Thresholds Definition
Command ===> _________________________________________________________________

Entry Type : System Defaults       PolicyDB Name   : USER_PDB
Entry Name : SYSTEM_DEFAULTS       Enterprise Name : USER_ENTERPRISE

Resource   : System wide defaults

Critical Number  . . . . 4_       (1 to 50)
Critical Interval  . . . 01:00    (hh:mm or hhmm, 00:01 to 24:00)

Frequent Number  . . . . __       (1 to 50)
Frequent Interval  . . . _____    (hh:mm or hhmm, 00:01 to 24:00)

Infrequent Number  . . . __       (1 to 50)
Infrequent Interval  . . _____    (hh:mm or hhmm, 00:01 to 24:00)
```

Figure 12. Example of Thresholds for System Defaults

For more details about this panel, see “THRESHOLDS Policy Item” on page 207.

When the number of occurrences of a situation reaches the infrequent and frequent error thresholds, SA z/OS notifies the operator and logs a message in the NetView log. When the number of occurrences of a situation reaches the critical threshold, SA z/OS stops to recover from that abend situation.

Once processing has stopped, operator intervention is required to restart it. This is because the critical error threshold is used as a trigger to stop SA z/OS from attempting to restart the application. After the operator has restarted the application, the thresholds will next be analyzed when the application next abnormally ends. If this occurs within the time interval specified for the critical threshold, it may result in the critical threshold being reached again. The error density will decrease with time, but this will not cause SA z/OS to restart the application.

For example, suppose an application has a critical error threshold of two per hour:

- The application will be restarted if it abnormally ends at 10:30.
- The application will be restarted if it abnormally ends again at 12:20.
- The application will not be restarted by SA z/OS if it abnormally ends again at 12:45. Operator intervention is required to restart the application.
- If an operator restarts the application at 13:05 and the application abnormally ends at 13:15, the error count will again reach two per hour and the application will not be restarted by SA z/OS.
- If the operator restarts the application at 15:00, the error count within the last hour will be zero again. The application will have to abnormally end twice within the hour in order to reach the critical error threshold again.

The primary use of error thresholds in SA z/OS automated operations is to track application abnormal ends and ensure that the abend-and-restart cycle does not continue indefinitely. You can also set error thresholds for events other than the occurrence of errors. For example, you can set thresholds to act as counters of specified events on your system.

Thresholds for Applications

Error threshold definitions for an application define how many abend errors can occur before a message is logged or restart processing is stopped. You define such a threshold using the THRESHOLDS policy item of an Application policy object. SA z/OS displays a panel similar to the one shown in Figure 12 on page 19.

If no error thresholds are set here, the application uses the thresholds that are defined in the Application Defaults policy object. Determine whether existing default threshold values are appropriate for this application. If they are not appropriate, create a unique threshold entry for the application.

Thresholds for Minor Resources

You can define thresholds for minor resources (such as messages or statuses) that can trigger automation for applications and MVSESA resources. You can do this using the MINOR RESOURCE THRES policy item of an APL or MVC policy object.

For example, a threshold for MVSESA.MVSDUMP specifies how often a full dump data set should be recovered in a particular time interval before the operator gets a message indicating that something has gone permanently wrong.

Thresholds can be set that let the operator know if certain errors are occurring infrequently, frequently or have reached a critical stage where the recovery process should be ended (to avoid endless loops). This is done by specifying how many times an error must happen in a certain time period for each error situation.

Automation Flags

One of the main tasks of customizing SA z/OS system operations involves setting automation flags. You can access Automation Flag Specification panels when defining Applications (individual applications or application classes), Application Defaults sets, MVS Component sets, System Defaults sets and MVSCOMP Defaults sets.

The Automation Flag Specification panels of the customization dialog define the cumulative effects of your automation flag settings, including:

- When automation is active
- Resources that automation applies to
- The types of automation mode you want SA z/OS to handle, such as starting and shutting down applications.
- Any exceptions to the defined automation
- Whether any automated operator function will operate in *log mode*, an SA z/OS option allowing a human operator to log the flow of an automated operation.

SA z/OS automation flags control whether automation occurs for a resource and, if so, when. Using the customization dialog, you can set default automation flags for all systems and applications or you can override automation flags for specific systems or applications.

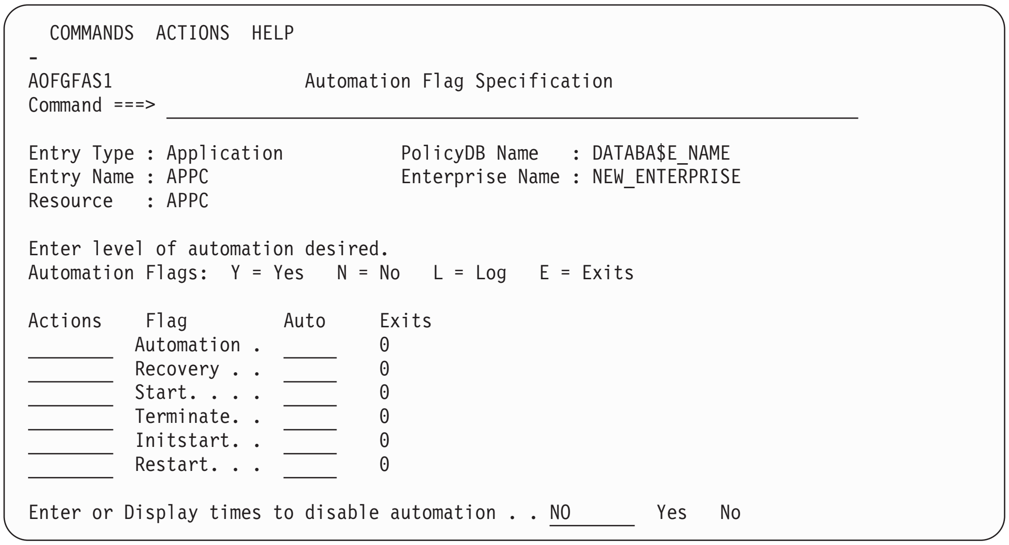
The following topics are discussed:

- “Types of Automation Flags” on page 21
- “Setting Automation Flags” on page 23
- “Automation Manager Flags” on page 24
- “Relationship Between Automation Flags and Resource Status” on page 24
- “Automation Flags and Minor Resource Flags” on page 25

For more information on how to use automation flags and when SA z/OS checks automation flags, refer to *IBM Tivoli System Automation for z/OS Customizing and Programming*.

Types of Automation Flags

Figure 13 shows an example of a Automation Flag Specification panel that is displayed when the AUTOMATION FLAGS policy item for an Application is selected (refer to Figure 108 on page 162).



```
  COMMANDS  ACTIONS  HELP
-
AOFGFAS1                 Automation Flag Specification
Command ===> _________________________________________________________________

Entry Type : Application          PolicyDB Name   : DATABA$E_NAME
Entry Name : APPC                 Enterprise Name : NEW_ENTERPRISE
Resource   : APPC

Enter level of automation desired.
Automation Flags:  Y = Yes   N = No   L = Log   E = Exits

Actions    Flag        Auto     Exits
________  Automation .  _____    0
________  Recovery . .  _____    0
________  Start. . . .  _____    0
________  Terminate. .  _____    0
________  Initstart. .  _____    0
________  Restart. . .  _____    0

Enter or Display times to disable automation . . NO______  Yes   No
```

Figure 13. Automation Flag Specification Panel

There is one global automation flag that controls all the other flags:

- **Automation**

  Automation is the global automation flag for resource automation. During automated operations, SA z/OS first checks this flag when determining whether automation is on or off for a resource and which operator functions are automated.

  The setting for this flag controls what responses are allowed, as follows:

  Y If the setting is YES, SA z/OS checks specific flag settings for that resource.

  N If the setting is NO, automation is turned off completely for the resource and SA z/OS does not check any other flags.

  L If the setting is LOG, the resource is running in LOG assist mode. This means that the normal automation actions that should be taken do not occur, but messages describing what would have happened are written to the NetView log.

  E If the setting is EXITS, there are one or more user-provided exit routines enabled for the resource.

The other flags that apply to each resource are:

- **Recovery**

  This flag determines if SA z/OS monitors a resource for predefined situations requiring recovery activity and takes necessary corrective actions when these situations occur. Some examples of situations requiring recovery activities include write-to-operator buffer (WTOBUF) shortages and full MVS system maintenance data sets, such as LOGREC, SYSLOG, and MVS dump, that require offloading. By automatically responding to error conditions, SA z/OS can prevent resource outages, thereby increasing system availability.

- **Start**

  This setting manages the startup process for an application by ensuring appropriate actions are taken, completely initializing the application, and bringing the application to an up or ready state (when the application issues an UP message, indicating that it is ready for use).

  For resources that have a startup process consisting of issuing several commands and handling several replies, you can use the Start flag to determine how much of the startup process is automated. For example, if you want to automate initial startup process for a resource but have an operator handle the rest of the startup process you can set the Initstart flag to YES, and the Start flag to NO. The resource start phase then requires operator intervention. You might want to set your flags like this for situations when you do not want to use the automated, predefined startup procedures but want to issue unique startup commands.

- **Terminate**

  This flag determines whether a shutdown command can be accepted and processed for an application. It also determines whether secondary automation will be processed if a manual shutdown is issued.

Note:

If you use the override parameter on INGREQ to force an automated shutdown when the Terminate flag is off, secondary automation will not occur as a result of the shutdown. This setting manages the shutdown process for an application by ensuring that actions are taken to stop the application in an orderly fashion, such as issuing shutdown commands and replies.

- **Initstart**

  This setting controls the initial startup process for an application following an SA z/OS initial status determination cycle. SA z/OS conducts status determination cycles when:

  – SA z/OS is initialized after an IPL.

  – SA z/OS is initialized when its host NetView is restarted.

  – The automation control file is reloaded.

  – The RESYNC command is issued.

  Those applications found to be inactive during the first SA z/OS initialization after an IPL are assigned an initial status of DOWN, if the status prior to the IPL and the various other options permit.

  Following SA z/OS initialization, those applications that have a status of DOWN will be started if the Initstart flag setting is YES. SA z/OS will ensure that the application is started in the order specified in automation policy and that any prerequisite applications are functional.

  For some resources, the activities controlled by the Initstart setting completely start up an application. That is, the MVS START command is sufficient to bring the resource to a STARTED state. For other resources, the startup process is more involved, with additional commands issued and replies handled. For such resources, you can use the Start automation flag to automate the rest of the startup process or indicate that an operator will handle the rest of the process.

- **Restart**

The setting of this flag controls the restarting of an application, either when it ends abnormally (an *abend* condition), when automation shuts it down, or when it shuts down outside the control of SA z/OS. By automatically responding to the application requiring a restart, SA z/OS automation decreases the time and system resources required to get an application up and running again.

Error thresholds influence whether SA z/OS restarts resources. During a condition requiring restart, SA z/OS checks whether the number of occurrences of the condition exceeds the critical threshold defined in the automation policy. If the critical threshold is exceeded, SA z/OS does not attempt restarting a resource. See “How SA z/OS Uses Error Thresholds” on page 17 for more information.

The **Exits** column displays the number of exits defined.

If you want to enable automation, but specify times when it should be disabled, you can enter YES in the input field at the bottom of the screen. Another panel will be displayed where you can specify one or more time ranges specifying the times when automation will be disabled.

Setting Automation Flags

You can set automation flags for:

- System Defaults (using their AUTOMATION FLAGS policy item)
- Application Defaults (using their AUTOMATION FLAGS policy item)
- MVSCOMP Defaults (using their AUTOMATION FLAGS policy item)
- Applications (using their AUTOMATION FLAGS policy item)
- MVS Components (using their MINOR RESOURCE FLAGS policy item)

Note:

For MVS components, only the Automation and Recovery flags apply.

Because flags have inherited values, it is not necessary to define them all explicitly. Specific resources need only have flags defined if they differ from the values inherited from the defaults.

A common step during automated handling of events is to check the automation flags to determine whether automation is on or off and whether automated handling of an event can continue.

The AOCQRY common routine performs this automation flag check. See *IBM Tivoli System Automation for z/OS Programmer’s Reference* for more information.

The benefit of this automation hierarchy is that you can customize SA z/OS to automate as many activities and resources as necessary in the z/OS environment. You can choose to have SA z/OS handle all phases of startup, monitoring, recovery, and shutdown for all resources or have a human operator control some phases for some resources. You can control all automation by setting just one flag (the Automation flag) on one panel and then setting exceptions. The simplest way to automate your resources is to use the Automation flag.

Periods that automation is not to apply for can be specified.

There are several commands to display automation flag settings, for example, DISPFLGS. The INGAUTO command dialog can be used to set or override automation flags.

SA z/OS provides an exit capability for automation flag processing. See *IBM Tivoli System Automation for z/OS Customizing and Programming* for more information.

Automation Manager Flags

There are two automation flags that are not set in the customization dialog, but are set either by an operator command or when the automation agents are initialized:

- The *global automation flag:*

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

  The purpose of this flag is to prevent (if flag is set to NO) or enable (YES) the starting or stopping of resources. This can be done for resources that are in systems that are currently inactive, for example, to prevent the startup of the resource at IPL time of the system.
- The *hold flag:*

  This flag can be set with the INGSET command but it is set automatically by SA z/OS in the following way:
  - If you initialize the automation agents you get the WTOR message AOF603D where you can select the NOSTART option. If you select this option, SA z/OS internally sets the hold flag for every resource, so that no resource (subsystem) is automatically started until this flag is overridden by an operator's explicit INGREQ request.

  The purpose of this flag is to enable the startup of manually selected subsystems instead of starting all subsystems at a time.

Relationship Between Automation Flags and Resource Status

From initialization through normal operation to shutdown, a resource can have several automation modes, such as being started, in an up state, abending, and shutting down. SA z/OS uses *resource states* to track these automation modes for monitored resources. A resource must have a particular status for certain automated actions to occur. The effect of these actions may, in turn, change the resource status from one value to another.

Understanding how SA z/OS typically uses resource states and automation flags is important when planning and writing your own automation procedures. For more details on automation flags, resource states, and their use in automation procedures, see *IBM Tivoli System Automation for z/OS Customizing and Programming* and *IBM Tivoli System Automation for z/OS User's Guide*.

SA z/OS provides an exit for processing state changes. Refer to *IBM Tivoli System Automation for z/OS Customizing and Programming* for more information.

Resource States and the Status Display Facility

SA z/OS also uses resource states in status information forwarded to the status display facility (SDF). This status information results in status changes being reflected on SDF panels.

Automation Flags and Minor Resource Flags

In addition to the automation flag settings for the entire application, you can set flags for minor resources (MINOR RESOURCE FLAGS policy item) that control responses to specific situations. Typical minor resources include message IDs and states.

For example, TSO is defined as application TSO. You have defined a reply to message IKT010D on the MESSAGES/USER DATA policy item of the Application policy object for TSO. You can change the shutdown process by:

- Turning the Terminate flag for TSO off. This means you have to shut down TSO manually.
- Turning the Terminate flag for minor resource IKT010D off. This allows SA z/OS to shut down TSO normally. When the IKT010D message is issued, shutdown processing will wait for a manual response.

Event and Trigger Support

This section informs you about the concepts of events and triggers and how you can use them. This information is useful to understand how to define these automation policies using the customization dialog.

How events and triggers are defined is described in sections:

- "Events Policy Object" on page 233
- "Triggers Policy Object" on page 238

Simply spoken, these facilities allow you to specify external conditions when certain applications on certain systems should either be automatically started or stopped.

Events are used to represent processes outside of automation. You can use the INGEVENT command to let SA z/OS know that an event has occurred. One or more events are part of a trigger condition. You can define a condition to be either a *startup condition* or a *shutdown condition*.

Events are used to decide whether the goal for a resource should be realized. Even if all dependency relationships and conditions are fulfilled, but a resource's startup condition is not fulfilled, because not all events of the startup condition are set, the resource will not be started.

An event can be in one of two states: set or unset. The state of an event is changed with an INGEVENT command. For more information on the INGEVENT command refer to *IBM Tivoli System Automation for z/OS Operator's Commands*. The state of an event is set by the user (or a procedure) and memorized throughout the sysplex. Each time, an event is set, SA z/OS checks all triggers defined in your environment or in your sysplex to find out whether applications need to be started or shut down according to the new conditions.

Triggers can be connected to applications or application groups. They act as inhibitors for the requested action. For an application to be started, its desired status must be AVAILABLE and its startup trigger must be satisfied. For an application to be stopped, its desired status must be UNAVAILABLE and its shutdown trigger must be satisfied.

If a trigger is connected to an application group then, if the trigger startup condition for the application group is not satisfied, none of its members can be started. Also, if the trigger shutdown condition is not satisfied, none of the group members can be stopped. So the trigger is kind of inherited to the members of the application group. This is also valid if the group contains nested groups. Then the trigger is also inherited to the members of the nested group.

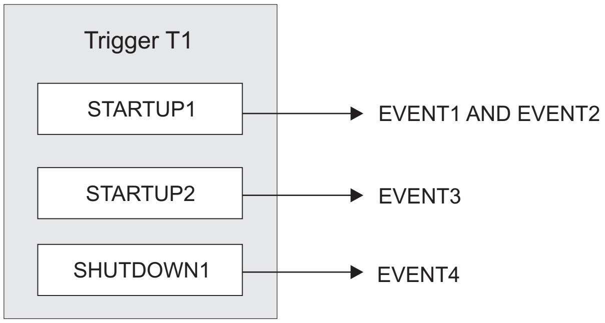


Figure 14. How a Trigger Uses Events

In Figure 14 the application connected to Trigger T1 would be started if either EVENT1 and EVENT2 are set or if EVENT3 is set, assuming the desired goal of the application is: available. The application would be stopped, if EVENT4 is set and the desired goal of the application is: unavailable.

Triggers are used to control the starting and stopping of applications in a single system or a sysplex. So the trigger must be linked to one or more applications and has a list of conditions attached to it.

A trigger's condition list consists of up to 10 startup conditions and up to 10 shutdown conditions, each having up to **20** events. Each condition is either a startup or a shutdown condition. All events within a condition must have the status *SET* to meet the condition. If more than one condition of the same type, *STARTUP* or *SHUTDOWN*, is specified, **only one** of that type has to be met to satisfy the trigger.

Triggers and Traffic Lights

A trigger is a kind of a traffic light that controls whether a running application should stop, or whether an application with goal *MAKEAVAILABLE* should really drive away from a traffic light that switched to green, just like a ready-to start resource may only run if the trigger's startup conditions are satisfied . You see that the goal, or the request to start a resource must be there in order that a trigger with fulfilled startup conditions has an effect, just as a green traffic light is of no importance if there are no cars around the street.

However, a trigger with fulfilled shutdown conditions does not work like a red traffic light. Instead, a shutdown trigger (the red traffic light) stops the application only if the application's goal is *MAKEUNAVAILABLE* (while a red traffic light stops a running car anyhow). And, an application with goal *MAKEUNAVAILABLE* is only stopped, if a connected shutdown trigger is set.

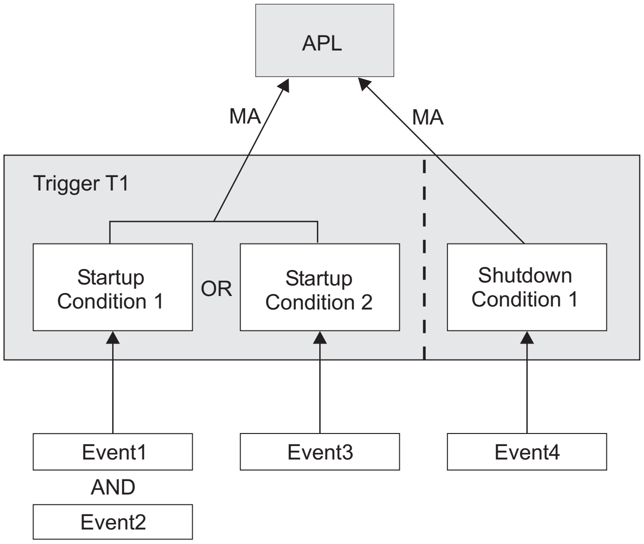


Figure 15. Triggers, Events and Conditions

Figure 15 shows another scenario, how triggers, events and conditions work together: for example, if EVENT1 and EVENT2 are set, Startup Condition 1 is true and Trigger T1 starts the application - provided that the application's goal is being available. If EVENT4 is set, Shutdown Condition 1 is true and Trigger T1 stops the application - provided that the application's goal is being unavailable.

A startup condition is satisfied, when all events of the condition are set. A shutdown condition is satisfied, when all its events are set.

Where no real dependency exists between the applications it is more appropriate to use events and triggers to control the STARTUP and SHUTDOWN.

Assigning System Automation Symbols (AOCCLONE)

You can assign up to 36 system automation symbols to each system (for example, &AOCCLONE., ~, &AOCCLONE1. through &AOCCLONE9., and &AOCCLONEA. through &AOCCLONEZ.). When you use a system automation symbol in a job name, each system can run with a unique job name while the application or subsystem shares a single automation policy across systems. The substitution can occur anywhere within the job name. Any of the 36 available system automation symbols can be used. You may specify more than one system automation symbol.

Note:
Be careful when cloning applications that are controlled by the Automatic Restart Manager. In most cases, the applications should be defined to Automatic Restart Manager as restart-in-place only. If you allow cross-system restarts for cloned applications, they may be restarted more than once on the same system.

The usage of system automation symbols is not limited to sysplexes. In addition to qualifying the job name, you can use system automation symbols for substitution anywhere within a defined literal, such as a shutdown command. The value of system automation symbols is also available for your automation through the

NetView AOFAOCCLONE common global variables (1 to 9, and A to Z). The AOFAOCCLONE and each AOFAOCCLONEn variable contains the value of the corresponding AOCCLONE variable.

Note:
The job name is the result of the substitution of any applicable &AOCCLONE. value and will be the name that is used for the whole automation of the application.

Using System Symbols and System Automation Symbols

You can specify system symbols as well as system automation symbols (AOCCLONE variables) in the fields in the customization dialog that are given in Table 2.

Table 2. System Symbols and System Automation Symbols

Field Name	Policy Item	Entry Type
ARM element name	APPLICATION INFO	APL
WLM resource name	APPLICATION INFO	APL
Job name	APPLICATION INFO	APL
JCL procedure name	APPLICATION INFO	APL
Startup parameter [1]	APPLICATION INFO STARTUP	APL
Command prefix	AUTOMATION	APL
IMS Control Region Name	IMS CONTROL	APL
IMS Subsystem ID	IMS CONTROL	APL
DB2 Subsystem name	DB2 CONTROL	APL
DB2 Active Log Dataset Name	DB2 CONTROL	APL
OPC Subsystem ID	OPC CONTROL	APL
CICS Major Node Name	CICS CONTROL	APL
CICS APPLID	CICS CONTROL	APL
Minor resource names [2]	MINOR RESOURCE FLAGS MINOR RESOURCE THRES	APL, MVC
Message ID	MESSAGES/USER DATA	APL, MVC
Command text [1]	MESSAGES/USER DATA (type CMD) STARTUP SHUTDOWN JES2/3 SPOOLFULL JES2/3 SPOOLSHORT JES3 MONITOR	APL
	MESSAGES/USER DATA	MVC
	TIMERS	TMR
Reply text [1]	MESSAGES/USER DATA (type REPLY) SHUTDOWN	APL
	MESSAGES/USER DATA	MVC
Code1/2/3, Value Returned [1]	MESSAGES/USER DATA (type CODE)	APL, MVC
Data [1]	MESSAGES/USER DATA (type USER)	APL, MVC

Table 2. System Symbols and System Automation Symbols (continued)

Field Name	Policy Item	Entry Type
Data [1]	E-T DATA	UET
Temporary Data Set HLQ	SYSPLEX	GRP
Monitoring Command [1]	USS CONTROL	APL
Command/Path [1]	USS CONTROL	APL
File Name [1]	USS CONTROL	APL
User ID	USS CONTROL	APL
Console Name	AUTOMATION CONSOLE	SYS
UNIX Installation Path [1]	AUTOMATION SETUP	SYS
Activate command [1]	MONITOR INFO	MTR
Deactivate command [1]	MONITOR INFO	MTR
Monitor command [1]	MONITOR INFO	MTR
Monitored Object [1]	MONITOR INFO	MTR
Host Name [1]	SOAP SERVER	NTW
SYMBOL1–SYMBOL9 [1]	APPLICATION SYMBOLS	APL

1. If you are using system automation symbols in fields that are not automatically translated to uppercase, these variables must be entered in uppercase.
2. The ending period of a system automation symbol for minor resource names must be followed by at least one character, for example, `DAT&AOCCLONE2.A`

Table 3 lists the fields in the customization dialog where you can only specify system symbols.

Table 3. System Symbols

Field Name	Policy Name
Automation Operators	Policy OPERATORS of entry type AOP
NetView Operators	Policy OPERATORS of entry type AOP
Console Name	Policy OPERATORS of entry type AOP

Table 4 lists the field in the customization dialog where you can only specify system automation symbols (AOCCLONE variables).

Table 4. System Automation Symbols

Field Name	Policy Name
Supporting Resource Name	Policy RELATIONSHIPS of entry type APL and APG

Nested Classes

Nested class support allows you to link one application class to another, thus enabling the specification of a class hierarchy. Links from a class can be made in two directions:

- *Downward* to multiple classes or instances, where the policy that is defined in the upper-level class is inherited by the classes or instances

- *Upward* to a single class, so that the lower-level application inherits the policy that is defined in the upper-level class

These classes can carry data on various levels, be nested, and an application can inherit data from a class chain. This can be particularly useful for policy databases with lots of applications of the same type, for example, large IMS installations with lots of IMS subsystems. You might define a top-level class with basic definitions for all IMS applications, and nested classes with more specific definitions for the various IMS subtypes, so that applications can inherit policy values from the top-level class and specific nested classes.

Class policies are inherited down to the instance level where any changes to the policy on that level override the inherited policy. Inherited policy data is displayed in a different color that you can change in the customization dialog Settings Menu (see "Changing Your Customization Dialog Environment" on page 391). Inherited data is displayed for linked instances for the following policies:

- "APPLICATION INFO Policy Item" on page 168
- "RELATIONSHIPS Policy Item" on page 179
- "MESSAGES/USER DATA Policy Item" on page 180
- "STARTUP Policy Item" on page 195
- "SHUTDOWN Policy Item" on page 199
- "THRESHOLDS Policy Item" on page 207
- "CICS CONTROL Policy Item" on page 218
- "DB2 CONTROL Policy Item" on page 221
- "IMS CONTROL Policy Item" on page 223
- "OPC CONTROL Policy Item" on page 228
- "USS CONTROL Policy Item" on page 229

Note that you can overwrite inherited data for an instance with the *same* value for the following policy items:

- APPLICATION INFO
- THRESHOLDS
- CICS CONTROL
- DB2 CONTROL
- IMS CONTROL
- OPC CONTROL
- USS CONTROL

Example: If an application instance is linked to a class with a job type of TRANSIENT and you change the Transient Rerun value, you also need to specify the Job Type TRANSIENT on the instance even though it has already been inherited from the class.

Class Hierarchy

Classes can be linked to each other and to instances according to the following rules:

- An application class can have one or more downward links to other classes or instances
- There can be only one upward link from an application class to any particular class
- There is no limit for the number of supported class levels

Figure 16 shows example links that are allowed or forbidden between classes and instances.

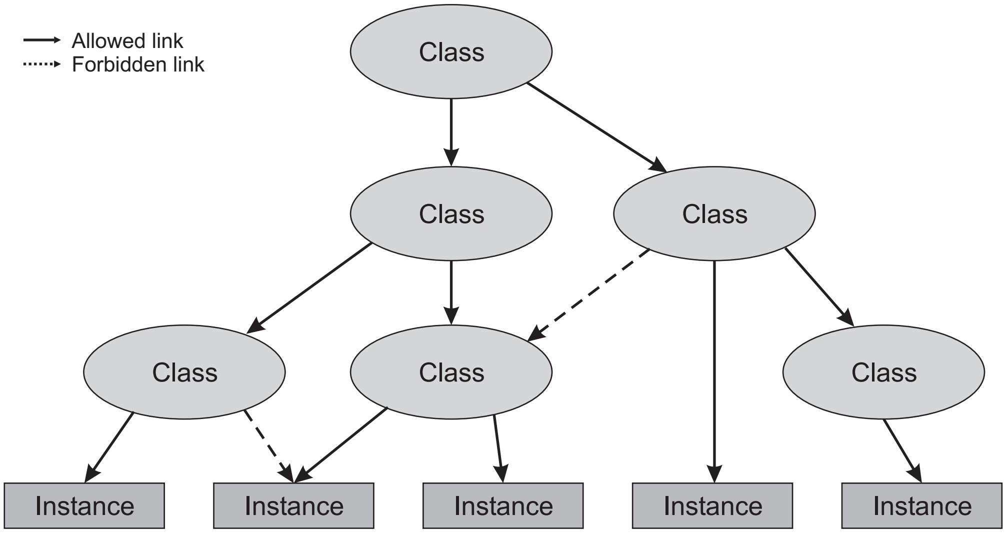


Figure 16. Nested Class Links

Inheritance and Defaulting

Figure 17 shows how inheritance from application classes relates to defaulting from the defaults policy items.

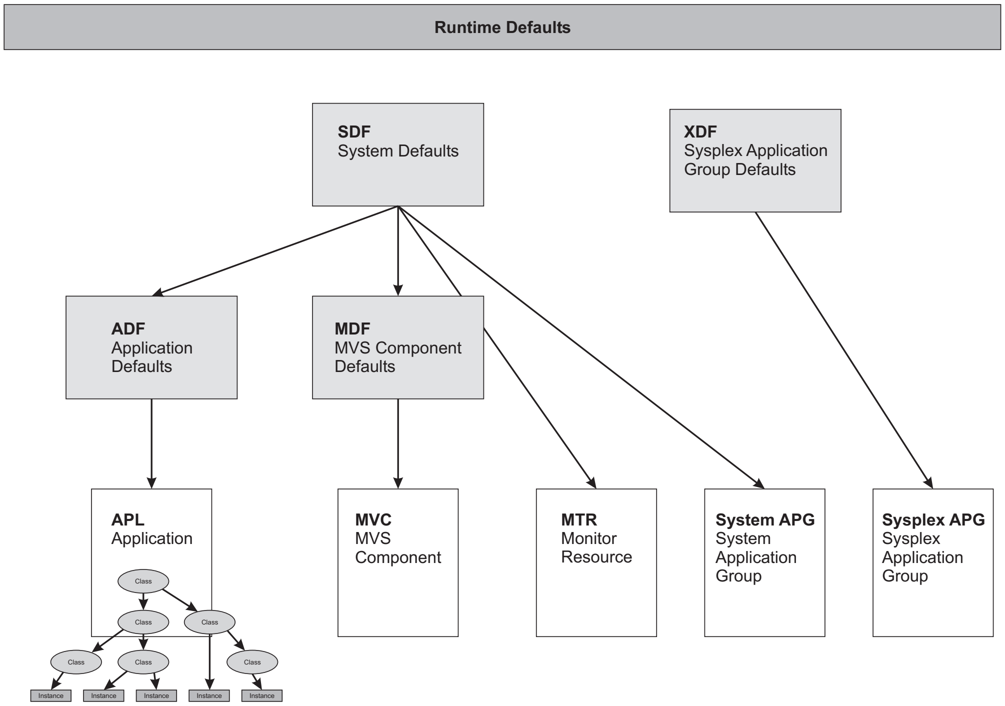


Figure 17. Inheritance from Classes and Defaulting

The data that is defined in Defaults policies and for application classes is handled differently. If class data is modified and built, a configuration refresh updates the applications that are linked to the modified classes. However, if defaults are modified and built, a configuration refresh updates only the global variables for the modified entries but not for the affected objects, for example, applications.

Class/Instance Link Considerations

You should be aware of the following when making links between classes and instances:

- Only applications with either the same type or no type specification can be in a link chain
- Type-specific policies are *only* available if the type is specified explicitly rather than inherited

For example, the CICS CONTROL policy item is available for an application of type CICS. It is not available, however, for an application that has no type specification but is linked to a class of type CICS.

Thus, if an application without a type is linked to a class with a type specification, you should be aware that you can only specify type-specific data for the class, but it will be inherited by the instance.

Furthermore, if you need to specify type-specific data at the application level, you must specify the type for the application. For example, some of the fields for applications of type USS allow mixed case. An application without a type specification does not allow mixed-case data, but it could inherit such data from a class of type USS.

Chapter 3. Using the Customization Dialog

The SA z/OS customization dialog runs under Interactive System Productivity Facility (ISPF) Dialog Manager, which operates under Time Sharing Option/Extended (TSO/E). For information on installing the customization dialog and sample policy databases on the focal point system, refer to *IBM Tivoli System Automation for z/OS Planning and Installation*.

The customization dialog provides panels to create and work with SA z/OS automation policy definitions. The task of defining and building automation policy is performed on the SA z/OS configuration focal point system from where it is distributed to the systems and sysplexes that you want to automate (see also Figure 259 on page 346).

The combined definition of all the resources is called an *enterprise definition* or also an *automation configuration*. SA z/OS enterprise definition data entered in the customization dialog is stored in an z/OS data set called a *policy database*. You can define more than one enterprise, each with its own policy database, or you can store the automation policy of an enterprise in more than one policy database, however, the policy for all systems of a sysplex must be in the same policy database.

Your first step in defining automation policy will be to create a new policy database, using one of the sample databases as a model. SA z/OS provides sample policy databases containing sample enterprise definitions. These are described in “Using SA z/OS Sample Policies” on page 323.

To do this, you should create an empty policy database using the *EMPTY sample database, and then use the *IMPORT* function to import policy definitions from either the other sample databases that are supplied with SA z/OS, or from your own policy databases. For more details about importing policy databases, see “Importing Policy Database Data” on page 352. You should note that this is now the preferred method for creating new policy databases.

You can create many different SA z/OS enterprise definitions. However, SA z/OS uses only one enterprise definition at a time. Each SA z/OS enterprise definition is stored in a different policy database. The customization dialog enables you to edit, copy, delete, and build SA z/OS enterprise definitions after you create them.

You select entry types from the list shown on the Entry Type Selection panel (Figure 2 on page 7) and certain objects from these entry types to specify the way enterprise resources are to be handled by SA z/OS when predefined conditions are met. You can link automation policies to individual systems as required to meet your automation needs.

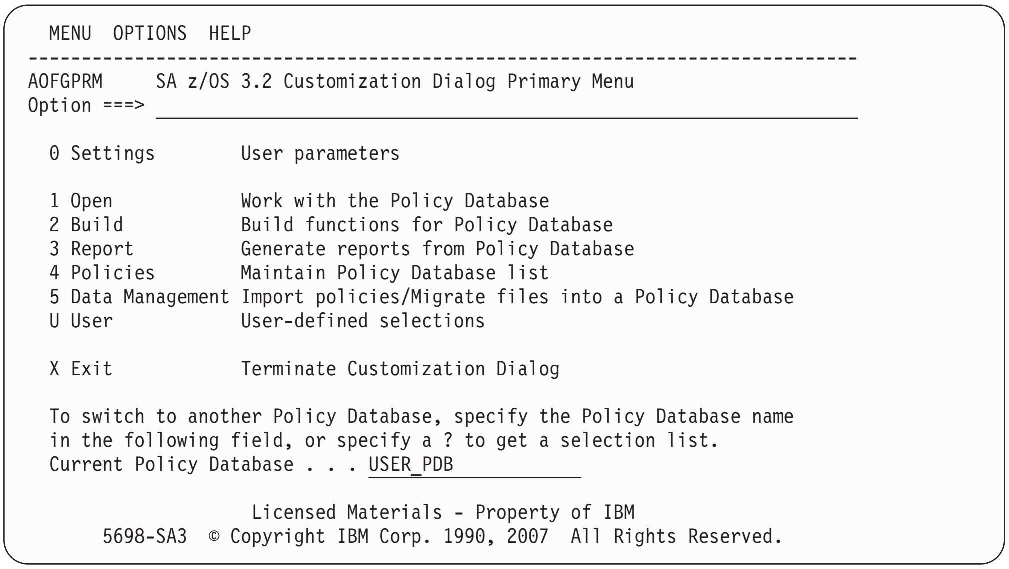
You can monitor systems and resources using the NMC workstation that are defined to SA z/OS even if you have not defined special automation policy for these resources. Conversely, you may automate your enterprise without monitoring it using the NMC workstation. If you want to use SA z/OS for automation only you must still use the customization dialog to create policy for the resources you want to automate.

How to Start the Customization Dialog

To start the customization dialog:

1. Log on to TSO/E using a logon procedure that will make the customization dialog available to you. Refer to *IBM Tivoli System Automation for z/OS Planning and Installation* for instructions on how to create an appropriate logon procedure.
2. Select the customization dialog from the ISPF menu that it has been installed on. SA z/OS displays the Customization Dialog Primary Menu as shown in Figure 18.

   Note: The installation can use the keyword INITSEL for the startup exec INGDLG to automatically provide an initial selection on the Customization Dialog Primary Menu. If this keyword is specified, you will not see the Customization Dialog Primary Menu as the first panel when invoking the customization dialog, but for example, the Entry Name Selection panel for a frequently used entry type.



```
  MENU  OPTIONS  HELP
------------------------------------------------------------------------------
AOFGPRM     SA z/OS 3.2 Customization Dialog Primary Menu
Option ===> __________________________________________________________________

  0 Settings        User parameters

  1 Open            Work with the Policy Database
  2 Build           Build functions for Policy Database
  3 Report          Generate reports from Policy Database
  4 Policies        Maintain Policy Database list
  5 Data Management Import policies/Migrate files into a Policy Database
  U User            User-defined selections

  X Exit            Terminate Customization Dialog

  To switch to another Policy Database, specify the Policy Database name
  in the following field, or specify a ? to get a selection list.
  Current Policy Database . . . USER_PDB____________

                     Licensed Materials - Property of IBM
       5698-SA3  © Copyright IBM Corp. 1990, 2007  All Rights Reserved.
```

Figure 18. Customization Dialog Primary Menu

To start the customization dialog for a policy database of your choice, you can:

- Select option *1 Open* and type the name of your desired policy database into the **Current Policy Data Base** field (if not filled in from a previous invocation). This will lead you directly to the Entry Type Selection panel shown in Figure 2 on page 7.
- Type a '?' in the **Current Policy Data Base** field to get a selection of available policy databases in the Policy Database Selection panel as shown in Figure 244 on page 319. From this panel then you select your desired policy database.
- You can also reach the Policy Database Selection panel by selecting option *4 Policies*.

If there are no policy databases displayed when you start the customization dialog, you must either create a policy database (see "Creating a New Policy Database" on page 319) or add an existing policy database to your display list.

If other users want to view the policy database you have just created, they must add it to their PolicyDB list using the ADD command.

How to Navigate in the Customization Dialog

Many characteristics of the customization dialog are ISPF facilities. See *ISPF User's Guide* for details. You need to know the following characteristics of the customization dialog:

- For panels that display a list that you are required to select one or more entries from and specify an action against that entry, you can:
  - Enter "*action entry name*" on the command line, for example, `OPEN` *`entry name`*
  - Enter the action in the **Action** column next to the appropriate entry, for example, enter OPEN in the **Action** column next to entry name.
  - Select the entry by entering S or SELECT in the **Action** column next to the entry. Move the cursor to the ACTIONS menu bar entry, press Enter, and enter the number corresponding to the action you want to take. For example, place an S in the **Action** column next to "entry name" and select 1 (OPEN) from the Action menu.

    The action most likely to be used on a certain panel is always the first action listed in the ACTIONS pull-down menu. When you enter S or SELECT in the **Action** column and press Enter, this default action is used. For example, entering `S` under **Action** for "MY_ENTERPRISE" in the Policy Database Selection panel opens the MY_ENTERPRISE database.

  Some panels display a list that you can select only one entry from. If you attempt to select more than one entry from such a list, only one selection is accepted. If one entry is already selected and you select another one, this new selection becomes active and the previous selection is removed.

  From a list of policy items, you can select multiple items. Then they are processed sequentially. Pressing the END (PF3) key takes you to the next selected item to be processed without returning to the selection panel.
- The COMMANDS pull-down menu lists all commands that can be entered at the command line.
- The ACTIONS pull-down menu lists all actions that can be entered in the **Action** column for panels displaying entry lists.
- The VIEW pull-down menu lists the different levels of detail that can be displayed for the entries in an entry list. For example, in many entry lists you can display just the entry name and short description, or you can also display a detailed description for each entry.

  Sample panels shown in this documentation may look different on your system depending on the selected view.
- On data entry panels, the END (PF3) key verifies input, saves the data, and exits the current panel.
- Pressing the Enter key verifies input and redisplays the current panel.
- In certain cases a long message is displayed in a pop-up window on the panel. If this pop-up window covers required input areas you can move it. For more information see the section "Movable Pop-Ups" in *z/OS ISPF Dialog Developer's Guide and Reference*.

How to Use an Entry Type as a Fast Path

If the policy database that you want to work with, is already displayed in the field **Current Policy Data Base** of the Customization Dialog Primary Menu, you can use the entry type abbreviation as a fast path:

- You can directly type the code for the entry type that you want to work with. For example, to reach the Entry Name Selection panel for Applications, just type:

  ```
  APL
  ```
- If you want to start with defining a new application group for example, just type:

  ```
  APG; NEW application group_name
  ```
- From every panel within the customization dialog, if you want to reach a certain entry type and you already know the name of the object that you want to edit, you can type, for example, to reach the APPC application:

  ```
  =APL; S APPC
  ```

  or to reach application group *CICS_APG*:

  ```
  =APG; S CICS_APG
  ```

  or to just reach the Entry Name Selection panel for Applications:

  ```
  =APL
  ```

How to Use a Policy Item as a Fast Path

The Entry Name Selection panel has a special feature that allows you to jump directly to a particular policy item for an entry (skipping the Policy Selection panel for that entry). If you enter an abbreviation for a policy item in the Action column of an entry, SA z/OS immediately takes you to the policy item that best matches what you have typed. For example, if you are in entry type APL, entering `I` takes you to the Application Information policy item, and entering `LC` takes you to the Link Instance to Class policy item.

There are no predefined abbreviations for policy items; SA z/OS finds the best match for any input, for example, the beginning of policies or of words, and significant letters. This also ensures that abbreviations work for user-defined policy items.

Note the following special cases:

- Characters reserved for actions will not be matched with policy items. For example, `S` is reserved for action “select”. Thus it will not be matched with policy items (such as “SERVICE PERIOD”).
- If there is no best match, you will be taken to the Policy Selection panel, with an error message.

Concurrent User Access

Because you may have separate administrators for different areas (for example, JES or VTAM) who should be able to do their updates independently, the customization dialog allows concurrent update for entry types SYS, APL, APG, and MTR. For other entry types, only one user at a time can access a policy database. See also Appendix A, “Concurrent Customization by Multiple Users,” on page 387 for further information.

Because you may have separate administrators for different areas (for example, JES or VTAM) who should be able to do their updates independently, the customization dialog allows concurrent update for entry types SYS, APL, APG, MTR, and UET. For other entry types, only one user at a time can access a policy database. See also Appendix A, “Concurrent Customization by Multiple Users,” on page 387 for further information.

How to Cancel Input on a Panel

To cancel your input on the current panel, type CANCEL or C at the Command prompt. CANCEL can be used to discard data entered on the panel, whether you have pressed the Enter key. However, once the message

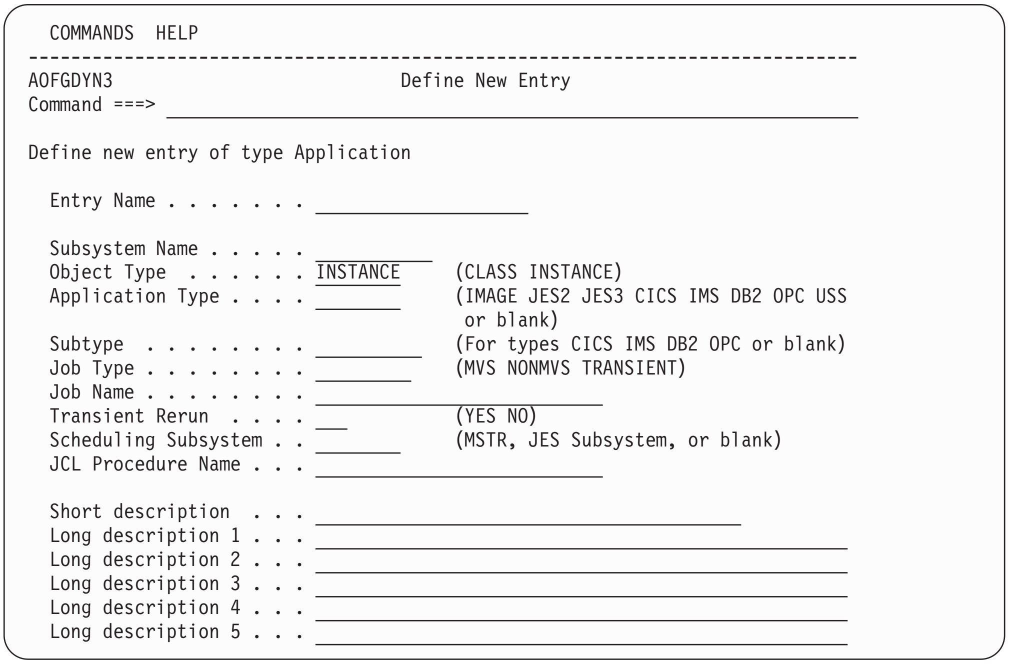
```
POLICY SAVED
```

is displayed when leaving a panel, the data entered has been saved and cannot be canceled. To delete input after this message is displayed, you must go back to that panel and type over existing values.

How to Create New Policy Objects

The process of creating a new policy object is the same for all types of policy objects, as follows:

1. From the Entry Type Selection panel, you select the desired policy object's entry type. For example, if you want to create a new Application object, enter option *6* or *APL* at the command line.
2. SA z/OS displays the Entry Name Selection panel for the selected entry type. You can either select action NEW from the COMMANDS pull-down or enter the command `new` *`entry_type_name`* at the panel's command line. The entry type name is optional for this command.
3. SA z/OS displays the Define New Entry panel (an example for defining a new Application is shown in Figure 19) for the selected entry type, where you now specify the required information. See Chapter 5, "Policy Object Reference," on page 91 for more information about how to create new policy objects.



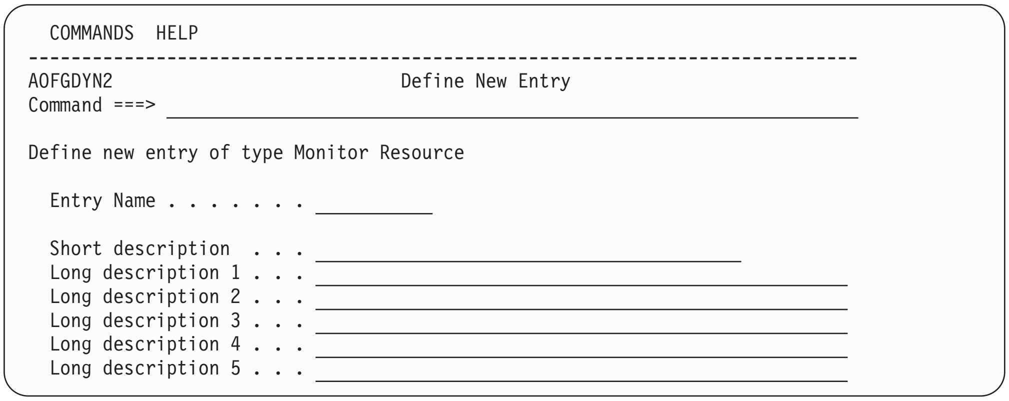
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN3                           Define New Entry
Command ===> _________________________________________________________________

Define new entry of type Application

  Entry Name . . . . . . . ____________________

  Subsystem Name . . . . . ___________
  Object Type  . . . . . . INSTANCE     (CLASS INSTANCE)
  Application Type . . . . ________     (IMAGE JES2 JES3 CICS IMS DB2 OPC USS
                                         or blank)
  Subtype  . . . . . . . . __________   (For types CICS IMS DB2 OPC or blank)
  Job Type . . . . . . . . _________    (MVS NONMVS TRANSIENT)
  Job Name . . . . . . . . ___________________________
  Transient Rerun  . . . . ___          (YES NO)
  Scheduling Subsystem . . ________     (MSTR, JES Subsystem, or blank)
  JCL Procedure Name . . . ___________________________

  Short description  . . . ________________________________________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 19. Define New Entry Panel for an Application

4. If you specify an entry name and press Enter, the subsystem name and job name are defaulted to the entry name, if possible. This is also true for all other entry types where the entry name can be used as a default for further qualifications

How to Create New Policy Objects

Throughout Chapter 5, "Policy Object Reference," on page 91, if there is no extra information given about creating a particular policy object, it is created using a generic panel, as shown in Figure 20.

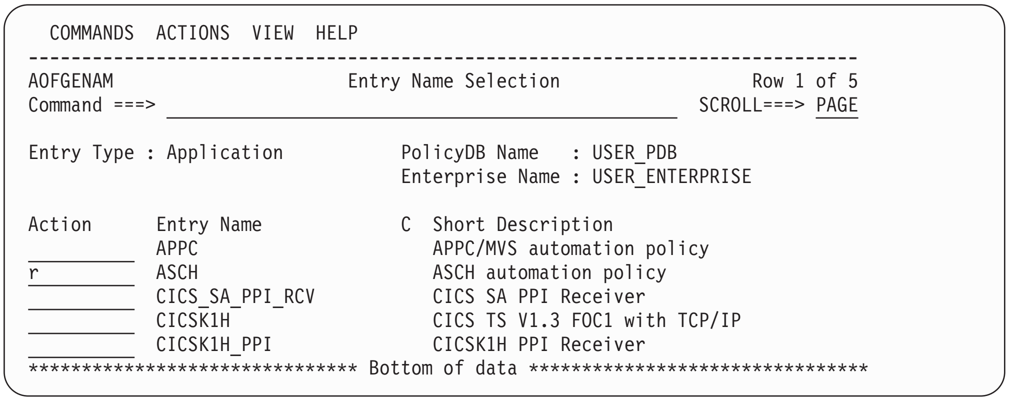


```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN2                            Define New Entry
Command ===> _________________________________________________________________

Define new entry of type Monitor Resource

  Entry Name . . . . . . . ___________

  Short description  . . . ________________________________________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 20. Sample Generic Define New Entry Panel

How to Rename a Policy Object

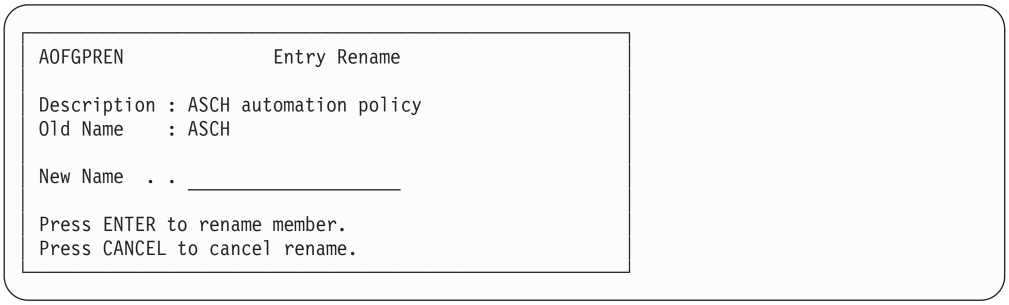
The process of renaming an existing policy object is the same for all types of policy object. From the Entry Name Selection panel, type `r` or `rename` in the **Action** field in front of the entry you want to rename, as shown in Figure 21.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGENAM                       Entry Name Selection                Row 1 of 5
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application            PolicyDB Name   : USER_PDB
                                    Enterprise Name : USER_ENTERPRISE

Action      Entry Name              C  Short Description
__________  APPC                       APPC/MVS automation policy
r_________  ASCH                       ASCH automation policy
__________  CICS_SA_PPI_RCV            CICS SA PPI Receiver
__________  CICSK1H                    CICS TS V1.3 FOC1 with TCP/IP
__________  CICSK1H_PPI                CICSK1H PPI Receiver
******************************* Bottom of data ********************************
```

Figure 21. Entry Name Selection Panel: Rename Action

The sample screen shows a list of applications with command `r` in the action field of entry name `ASCH`. Pressing the Enter key will display a rename popup panel where you can enter the new entry name, as shown in Figure 22.



```
AOFGPREN              Entry Rename

Description : ASCH automation policy
Old Name    : ASCH

New Name  . . ____________________

Press ENTER to rename member.
Press CANCEL to cancel rename.
```

Figure 22. Entry Name Rename Popup Panel

Pressing Enter will start the renaming process; if the name you entered already exists or is invalid you will be required to enter a different name. After completion, the newly sorted Entry Name Selection panel will be displayed with the renamed entry as the top row. If there are multiple rename requests, the last renamed entry is shown as top row.

You can leave the rename popup panel using END or CANCEL without renaming the policy object.

Note:
To modify the Enterprise name, edit the policy database and then simply overtype the name, see “Specifying Settings for Policy Databases” on page 324.

How to Delete Policy Objects

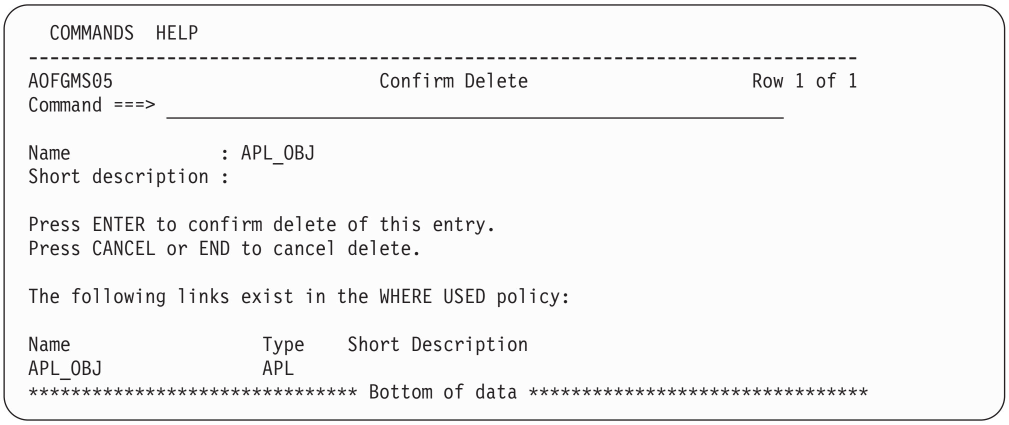
There are two ways to delete policy objects; you can:

- Delete a single policy object
- Delete any number of policy objects in one go (bulk deletion)

Note that if a *class* is selected for deletion and there are objects linked to it that inherit data, the confirmation panel is displayed. If a class is deleted that has only upward links, no confirmation panel is shown. These links are removed without further notification because there is no object left that loses data.

How to Delete a Single Policy Object

To delete a single policy object use action d against the policy object that you want to delete. This displays a confirmation panel; the example shown in Figure 23 lists any links that exist to the policy object.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMS05                         Confirm Delete                   Row 1 of 1
Command ===>

Name             : APL_OBJ
Short description :

Press ENTER to confirm delete of this entry.
Press CANCEL or END to cancel delete.

The following links exist in the WHERE USED policy:

Name                 Type    Short Description
APL_OBJ              APL
****************************** Bottom of data ********************************
```

Figure 23. Delete Confirmation Panel for a Single Policy Object

If the policy object to be deleted has any policy objects selected in its WHERE USED policy item (as reported in the confirmation panel), deleting it will automatically remove it from the corresponding policy items in those other policy objects (see Figure 26 on page 44 for the corresponding policy items).

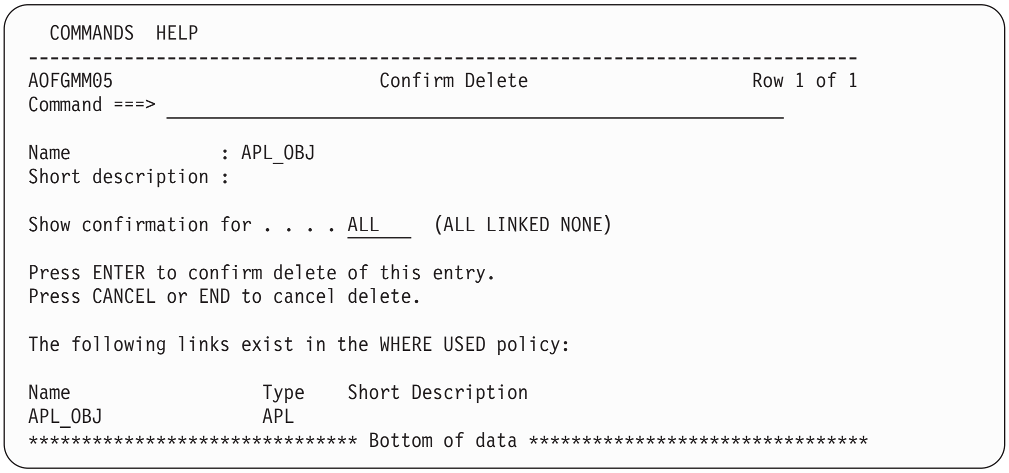
Links between APL policy objects of object type CLASS and object type INSTANCE are treated analogously.

If a class is selected for deletion and there are objects linked to it that inherit data, a confirmation panel is displayed. However, if a class is deleted that has only upward links, a confirmation panel is not displayed. These upward links are removed without further notification because there are no objects left that lose the data.

How to Delete Several Policy Objects in One Go (Bulk Deletion)

Only policy objects of the same entry type can be deleted in one go.

To delete several policy objects in one go, type d in the action column against each of the policy objects that you want to delete and press Enter. This displays the confirmation panel shown in Figure 24.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMM05                         Confirm Delete                    Row 1 of 1
Command ===> __________________________________________________________

Name              : APL_OBJ
Short description :

Show confirmation for . . . . ALL___  (ALL LINKED NONE)

Press ENTER to confirm delete of this entry.
Press CANCEL or END to cancel delete.

The following links exist in the WHERE USED policy:

Name                  Type    Short Description
APL_OBJ               APL
******************************* Bottom of data ********************************
```

Figure 24. Confirmation Panel for Bulk Deletion with WHERE USED Links

You can use the **Show confirmation for** field to set how confirmation of the bulk deletion is handled:

ALL
Only the policy object that is named on the confirmation panel will be deleted. The confirmation panel will then be redisplayed with the name and links of the next policy object that you marked with d.

This will continue until you change the value in the **Show confirmation for** field or all the policy objects that you marked with d have been deleted.

LINKED
The policy object that is named on the confirmation panel will be deleted. SA z/OS then processes the rest of the policy objects that you marked with d in the following way:

- Any policy object that is *not* involved in links will be deleted without further confirmation.
- As soon as a policy object with links is encountered the confirmation panel is redisplayed with that policy object's name and links on it.

NONE
All the policy objects that you marked with d will be deleted at once, without further confirmation.

Note that when a policy object with links is deleted, the links are treated exactly as in "How to Delete a Single Policy Object" on page 39.

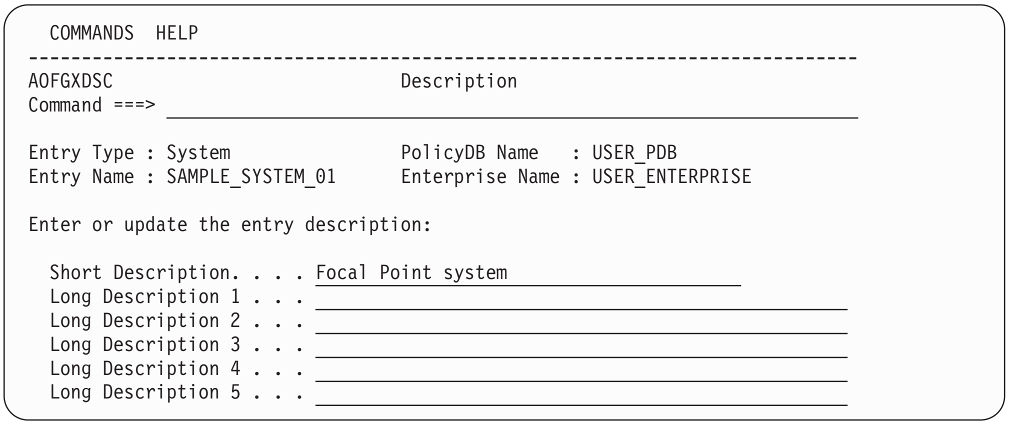
Common Policy Items

The following policy items are found on many policy definitions:

- DESCRIPTION
- COPY
- WHERE USED

DESCRIPTION Policy Item

Figure 25 shows an example of a Description policy item panel (in this case for the SYSTEM entry type)



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGXDSC                          Description
Command ===> _________________________________________________________________

Entry Type : System               PolicyDB Name   : USER_PDB
Entry Name : SAMPLE_SYSTEM_01     Enterprise Name : USER_ENTERPRISE

Enter or update the entry description:

  Short Description. . . . Focal Point system
  Long Description 1 . . . __________________________________________________
  Long Description 2 . . . __________________________________________________
  Long Description 3 . . . __________________________________________________
  Long Description 4 . . . __________________________________________________
  Long Description 5 . . . __________________________________________________
```

Figure 25. Sample Description Policy Item Panel

Both types of description (short and long) can be displayed in the customization dialog, depending on the VIEW option that you choose.

Short Description
: This is optional and can be up to forty characters in length. The short description is also used by the NMC workstation, DISPSTAT, and INGLIST.

Long Description
: This is optional and can consist of up to five lines of up to fifty characters in length. The extended description is part of the DISPINFO display for applications.

COPY Policy Item

When creating a new policy object, you can use this policy item to copy policy information from an existing object to the created object. This function displays a list of all the other entries of the same entry type in the SA z/OS enterprise that are eligible to be used as the source of a copy function.

Copy Process

The following outlines the process of using an object's COPY policy item when creating a new object.

1. Select an entry type from the Entry Type Selection panel. The Entry Name Selection panel is displayed.
2. Enter new (N) at the command line to create a new object. SA z/OS displays the Define New Entry panel for the policy object that you want create.
3. Enter information for the new object as prompted by its Define New Entry panel. Exit the panel (END command).

4. The Policy Selection panel for the new object is displayed. Select its COPY policy item.
5. A panel listing all the objects that are currently defined from the same entry type is displayed. Select the concrete policy object that you want to copy policy information from. Take note of the following:
   - Groups of type SYSPLEX can only be copied from groups of type SYSPLEX, and groups of type STANDARD can only be copied from groups of type STANDARD (see “Creating a New Group” on page 99).
   - For entry type SYS, copying is only possible for those entries that have the same operating system.
   - For entry type APL, copying is only possible for those entries that have the same application type.
   - Sysplex Application Groups can only be copied from Sysplex Application Groups, and System Application Groups can only be copied from System Application Groups (see “ApplicationGroup Policy Object” on page 134).
6. When the copy is complete, you return to the Policy Selection panel for the new object. A `Copy complete` message shows that the copy has been successful.

   If you copy groups or processors, you may receive the message `Partial copy complete`, which means that it has not been possible to copy all entries. The help function gives details about the data that was not copied.
7. Complete the new application definition by performing any necessary modifications to the information that has been copied.

Information that Is Copied

The following indicates what information is or is not copied by the COPY policy item:

1. All source object data is copied to the target object for all entry types with two exceptions:
   - Data that has to be entered during the ’NEW’ processing on the related Define New Entry panel for the target object is not copied from the source object.
   - The short and extended description that is also available on the Define New Entry panel is copied from the source object whenever this is not already specified for the target object.
2. All links to other objects down the hierarchy of objects are copied from the source to the target. For example, if you copy an application group with linked applications, the links of the source application group are copied to the target application group.
3. All links to other objects up the hierarchy of objects are NOT copied from the source to the target. For example, if you copy an application that is linked to an application group, the copied target application is NOT linked to that application group.
4. If you copy a class or instance with “upward” links (that is, links to a class), these links are copied. However, if you copy a class that has “downward” links, they are *not* copied.
5. Links to members of sysplex groups (that is, links to subgroups, systems and Sysplex Application Groups) are *not* copied, to avoid conflicts like having the same system defined in more than one sysplex.
6. Links from systems to processors are not copied to avoid conflicts having the same system defined in more than one processor.
7. Resource links of Application Groups are not copied.

WHERE USED Policy Item

Almost every entry type has the WHERE USED policy item.

Note:
The only entry types that do not have the WHERE USED policy item are ENT, GRP, APL classes, and PRO.

The WHERE USED policy item is effectively a container for *references to* existing policy objects. The container may be empty. To put a reference to a policy object in the container, *select* that policy object from within WHERE USED. SA z/OS automatically ensures that each reference in WHERE USED is matched by a reverse reference in an appropriate policy item of the other policy object (details further on in this section). Thus, WHERE USED actually contains one direction of *mutual* references. These mutual references are called “WHERE USED links” or “links”, for short.

What Policy Objects Can Be Linked?

Whether it is possible to have a link between two policy objects depends on their entry types. The entry types that SA z/OS supports are arranged in the hierarchy as shown in Figure 26 on page 44. The figure shows the various counterparts of WHERE USED. By following the arrows backwards, you can tell what information you will find in WHERE USED. That is, the entry type where a given policy object may be used.

Key to the figure:
Gray boxes in Figure 26 on page 44 represent entry types, and arrows represent policy items. When an arrow is attached to a white box inside a gray box (rather than to the gray box itself), the policy item that the arrow represents is only present on policy objects with the specific property named in the white box. An arrow named *z* from an entry type named *X* to an entry type named *Y* means:

- Every policy object in entry type *X* has a policy item named *z*. This policy item contains zero, one or more references to policy objects in entry type *Y*.
- Conversely, every policy object in entry type *Y* has the WHERE USED policy item. Any references in its WHERE USED policy item are to policy objects in entry type *X*.

With a few exceptions, the name of an arrow matches the name of the entry type that it points to (that is: the one that is *lower* in the hierarchy).

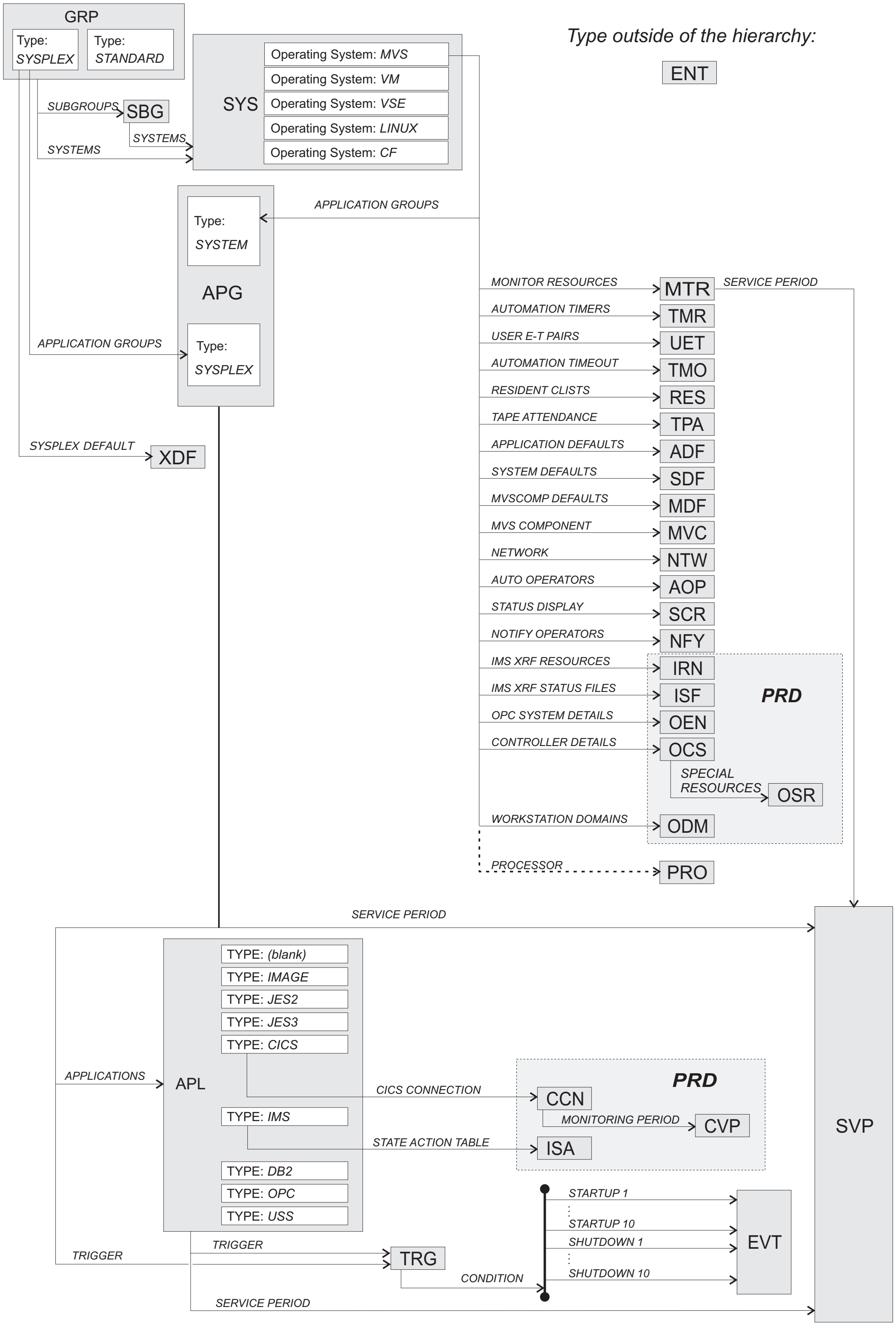


Figure 26. Hierarchy of Entry Types (Inverses of WHERE USED)

How to Create, View or Remove Links

Because links are *mutual* references, there are two alternative approaches to create, view or remove them.

Bottom-Up Approach:

Step 1. In the lower policy object (regarding the hierarchy in Figure 26 on page 44), open the WHERE USED policy item.

This displays a list of all the policy objects where the current one *may* be used. Those that have a status of SELECTED are the ones where the current one *is* actually used at present.

Step 2. To create a link, apply action “s” to the policy object in question.

Step 3. To remove a link, apply action “m” to the policy object in question.

Top-Down Approach:

Step 1. In the upper object (regarding the hierarchy in Figure 26 on page 44), open the policy item that takes you to the intended lower type, as shown in Figure 26 on page 44.

This displays a list of all the policy objects that the current one *may* use. Those that have a status of SELECTED are the ones that the current one *does* actually use at present.

Step 2. To create a link, apply action “s” to the policy object in question.

Step 3. To remove a link, apply action “m” to the policy object in question.

Chapter 4. Scenarios on How to Use the Customization Dialog

This chapter provides a series of scenarios on how to use the customization dialog to implement the appropriate automation policy:

This section discusses the following scenarios:
- "How to Work with Resources"
- "How to Use Schedules (Service Periods)" on page 71
- "How to Use Triggers and Events" on page 72
- "How to Use User E-T Pairs" on page 83
- "How to Implement Message Processing with User-Defined Data" on page 85

How to Work with Resources

This chapter contains a scenario on how to group resources for applications in a complex configuration. This scenario is based on the grouping scenario from "A Grouping Scenario" on page 12.

Some background information about resources will be given where required.

The complete process of grouping resources is divided into three steps:
- "Step 1: Creating the Policy Objects that Make Up the Sysplex"
- "Step 2: Creating the Applications" on page 49
- "Step 3: Creating the Application Groups" on page 50

Step 1: Creating the Policy Objects that Make Up the Sysplex

You start on the Entry Type Selection panel. It is assumed that the policy database is empty.

Creating a sysplex consists of:
- Creating the systems
- Creating the sysplex group
- Linking the systems to the sysplex group

Step 1.1: Creating the Systems

In this substep, you create the systems SYSTEM1, SYSTEM2 and SYSTEM3. You may want to read the information in "Creating a New System" on page 118 to see some sample customization dialog panels before you continue this scenario.

1. On the Entry Type Selection panel enter option 4 or SYS to get to the Entry Name Selection panel for systems, which currently displays an empty list.
2. Enter `NEW SYSTEM1` at the command line.

   The Define New Entry panel for systems is displayed; the **Name** field is already filled with `SYSTEM1`.
3. Enter `SYSTEM1` into the **MVS SYSNAME** field and press the PF3 (END) key twice to leave this panel.

Note:

To leave this panel, you have to press PF3 twice because the first PF3 prompts you for the ProcOps name, which is defaulted to the MVS SYSNAME if you do not change it.

Background:

What is a Resource?

The MVS SYSNAME name will be part of the resource names that have a system scope.

This name usually consists of three parts:

- An 11-character identifier
- The entry type that it was derived from
- The System that it runs on (that is, the MVS SYSNAME that is specified for the system)

For sysplex-wide resources (for example, Sysplex Application Groups), the system identifier is omitted.

Resources of type APL, APG, or MTR are automatically generated when applications, application groups, or monitor resources are linked to systems. Resources of other types are generated dynamically when the system operations configuration files are built.

Within the customization dialog, resources of type APL, APG, or MTR can be viewed and they can be linked via special policy items.

The Policy Selection panel for systems is displayed, showing a message indicating that the system is created.

4. Press PF3 or enter END to return to the Entry Name Selection panel showing SYSTEM1 in the list.
5. Repeat the previous steps to create SYSTEM2 and SYSTEM3. Now you have three systems in the list on the Entry Name Selection panel. Press PF3 to return to the Entry Type Selection panel.

Step 1.2: Creating the Sysplex

You may want to read the information in “Creating a New Group” on page 99 to see some sample customization dialog panels before you continue with the scenario.

1. On the Entry Type Selection panel, enter option 2 or GRP to go to the Entry Name Selection panel for groups which displays an empty list.
2. Enter `NEW SYSPLEX1` at the command line.

   The Define New Entry panel for groups is displayed; the **Name** field is already filled with `SYSPLEX1`.
3. Change the **Group Type** field entry to `SYSPLEX` because you want to create a sysplex group.

   Press PF3 to exit this panel. The Policy Selection panel for Groups is displayed, showing a message that the group is created.

4. Consider specifying *default definitions* for all your sysplex resources (that is, sysplex application groups) using the Sysplex Defaults entry type (XDF) and linking it to your sysplex group.
5. On the Policy Selection panel, select policy item SYSTEMS. The Systems for Group panel is displayed showing the three systems defined in "Step 1.1: Creating the Systems" on page 47.
6. Select all three systems with an 'S' and press PF3 twice.

   Now your environment contains a sysplex with three z/OS systems.

Background:

Ways to Build a Sysplex

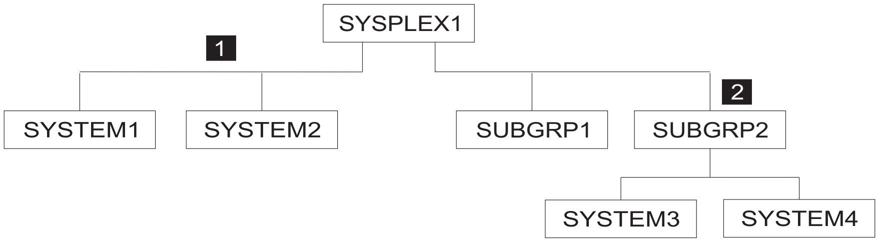
A sysplex for SA z/OS consists of z/OS systems within an XCF group with the same group ID. In the customization dialog such a sysplex **must** be built by creating a sysplex group and linking its systems.

This happens either ″bottom-up″ (from a system to a group, with the system's WHERE USED policy item) or ″top-down″ (from a group to a system, with the group's SYSTEM policy item).

For the ″bottom-up″ approach you must go into the SYS entry type, select all systems you want in the Sysplex, select each system's WHERE USED policy item where you get a list of sysplex groups and subgroups that you can link the system to.

You can also link groups with systems indirectly. This happens if you create SubGroup objects (SBG entry type) and link these subgroups ″bottom-up″ to groups AND ″top-down″ to Systems.

Sysplex-Groups usually consist of multiple systems; however, systems can only be linked to one sysplex group (either directly or indirectly).



The previous figure describes how you can create a sysplex:

1 SYSTEM1 and SYSTEM2 are linked directly to sysplex SYSPLEX1.

2 SYSTEM3 and SYSTEM4 are linked to subgroup SUBGRP2 and this subgroup is linked to SYSPLEX1.

Step 2: Creating the Applications

In this step, you create all required application subsystems.

You may want to read the information in "Creating a New Application" on page 157 to see some sample panels of the customization dialog before you continue with the scenario.

Step 2: Creating the Applications

1. On the Entry Type Selection panel enter option 6 or APL to get to the Entry Name Selection panel for applications, which is displayed with an empty list.
2. Enter `NEW VTAM` into the command line.

   The Define New Entry panel for applications is displayed; the **Name** field is already filled with `VTAM`.
3. Press Enter to have the **Subsystem Name** and **Jobname** fields filled with their default value. These fields are mandatory. Both of them default to the application's **Entry name** (see "Subsystem Name" on page 157 and "Job Name" on page 160 for more information on this defaulting mechanism). So, in this case, they default to VTAM.

> **Background:**
> *Application Resource Name*
>
> The Subsystem Name is used as first part of the application's resource name. It is recommended to use the same name for the application entry name and subsystem name for better identification of applications and resources that will be derived from applications.

4. Then press the PF3 to leave this panel. The Policy Selection panel is displayed, showing a message that the application is created.
5. On the Policy Selection panel for applications press PF3 again to return to the Entry Name Selection panel. It now shows VTAM in the list.
6. Repeat the previous steps for the rest of the applications using the values from Table 5.

Table 5. Entry Details for Applications in the Scenario

Entry Name	Application Type	Subtype
WBSRV (Web Server)	blank	–
DWH (Data Warehouse)	blank	–
CICSMAIN	CICS	TOR
CICSTOR	CICS	TOR
CICSFOR	CICS	FOR
CICSAOR1	CICS	AOR
CICSAOR2	CICS	AOR
CICSAOR3	CICS	AOR
DB2MAIN	DB2	–
DB2IRLM	DB2	–
TCPIP	blank	–

Now all subsystems (applications) are defined to your environment and the Entry Name Selection panel shows 12 applications.

Step 3: Creating the Application Groups

In this step, you create all required application groups and define the behavior of their members.

According to the example from "A Grouping Scenario" on page 12, the groups should run on the following systems:

- WBSRVAPG should run on SYSTEM1.
- DWHAPG should run on SYSTEM2 and alternatively on SYSTEM3.
- DB2 should run on SYSTEM2 and SYSTEM3.
- NET should run in the entire sysplex.
- CICS should run on SYSTEM2 and on SYSTEM3.

Background:

Sysplex Application Groups and System Application Groups

The application group type (SYSTEM or SYSPLEX) sets the boundaries for the automation of the application group and its members.

- Sysplex Application Groups may be located on multiple systems within one or more sysplexes.
- System Application Groups are located on single systems.

Use Sysplex Application Groups for DWHAPG, NET, and CICS. System Application Groups will be used for WBSRVAPG and DB2.

Step 3.1: Create Sysplex Application Group CICSAOR

You start with the creation of CICSAOR because CICSAOR contains only applications (CICSAOR1, CICSAOR2, CICSAOR3) that already have been defined. Then you create the application group with a nature of BASIC.

You may want to read the information in "Creating a New ApplicationGroup" on page 134 to see some sample panels of the customization dialog before you continue this scenario.

1. On the Entry Type Selection panel enter option 5 or APG . The Entry Name Selection panel is displayed with an empty list.
2. On the Entry Name Selection panel enter `NEW CICSAOR` into the command line. The Define New Entry panel for application groups is displayed; the **Name** field is already filled with `CICSAOR`.
3. Check that the **Application Group Type** field contains `SYSPLEX`.
4. Enter `BASIC` in the **Nature** field.

Background:

Nature of an Application group

SA z/OS knows three natures:

BASIC
In a BASIC application group *all* of its components must be available before the group is considered to be available.

MOVE
In a MOVE application group exactly one of its components must be available before the Group is considered to be available. In MOVE groups you can specify alternative components to start if its primary component fails to start.

SERVER
In a SERVER application group any number of components can be specified that must be available before the group is considered to be available. In SERVER groups you can specify what should happen if its components fails to start.

The nature of an application group can be changed later on. However be careful with such a change, because the role of the application group will change too (and definitions made for example for a MOVE group may not be appropriate for a SERVER group).

In the example, the application group consists of three applications (CICSAOR1, CICSAOR2, and CICSAOR3) that should all be available. You should therefore decide to take a BASIC Group.

5. Enter `CICSAOR` into the **Automation Name** field.

Background:

Application group Resource name

The Automation Name will be used as the first part of this application group's resource name. It is recommended to use the same name as the application group entry name and the Automation name for better identification of application groups and resources that will be derived from them.

If no Automation Name is specified, no resource will be generated for the application group. Nevertheless, the application resources for all applications linked to this application group are created and will become available for automation.

6. Press PF3 to exit the panel. The Policy Selection panel for Groups is displayed showing a message that the application group is created.
7. On the Policy Selection panel select the WHERE USED policy item. The Where Used panel is displayed with sysplex group SYSPLEX1.
8. Select the sysplex group and press PF3. For a moment, you can see the Command Progress Display popup window telling you that the resource has

been created. This popup is just a quick information that will disappear automatically. You can see the generated resources under the corresponding policy item of the application group.

Finally, you are back on the Policy Selection panel for CICSAOR.

Background:

Generating Sysplex Application Group Resources

A Sysplex Application Group resource is created automatically if:

- there is an Automation Name for the application group, and
- there is a link to a sysplex group.

Only one resource will be created (not one resource per system) and the resource name has no system identifier.

CICSAOR has got an Automation Name; therefore a resource for this application group will be generated.

The current configuration with application group CICSAOR is now:

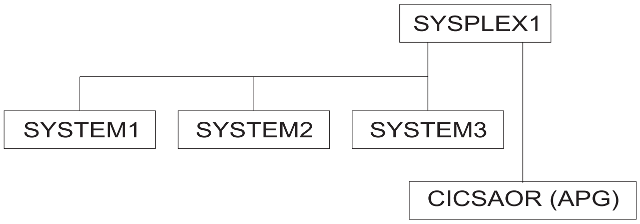


Figure 27. Current Configuration with Sysplex Application Group CICSAOR

9. On the Policy Selection panel for CICSAOR select the APPLICATIONS policy item.

   The Applications for ApplicationGroup panel will be displayed with all twelve applications.

10. Select all of

    ```
    CICSAOR1
    CICSAOR2
    CICSAOR3
    ```

    and press PF3 (END).

    You are now back on the Policy Selection panel for CICSAOR.

Background:
Generating Application Resources

Application Resources are generated when Applications are linked to Systems. This now is the case for CICSAOR1, CICSAOR2, CICSAOR3: all now are linked to CICSAOR, and indirectly via sysplex SYSPLEX1 they are linked to SYSTEM1, SYSTEM2 and SYSTEM3.

For each linked system, one resource is generated.

Again, you can see the Command Progress Display popup window telling you that the resources have been created. For example, for CICSAOR1, the following resources have been generated:

```
CICSAOR1/APL/SYSTEM1
CICSAOR1/APL/SYSTEM2
CICSAOR1/APL/SYSTEM3
```

For each application three resources are generated and linked to the application group resource.

Note: These Application Resources are deleted when:
- either the application is deleted,
- or the application group CICSAOR is deleted,
- or if other parts of the link chain APL-APG-GRP-SYS are deleted or de-linked (that is, if SYSPLEX1 or Systems SYSTEM1...3 or links between them are deleted).

The current links are:

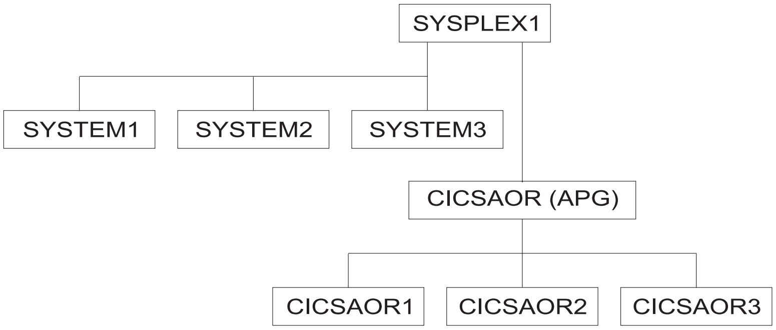
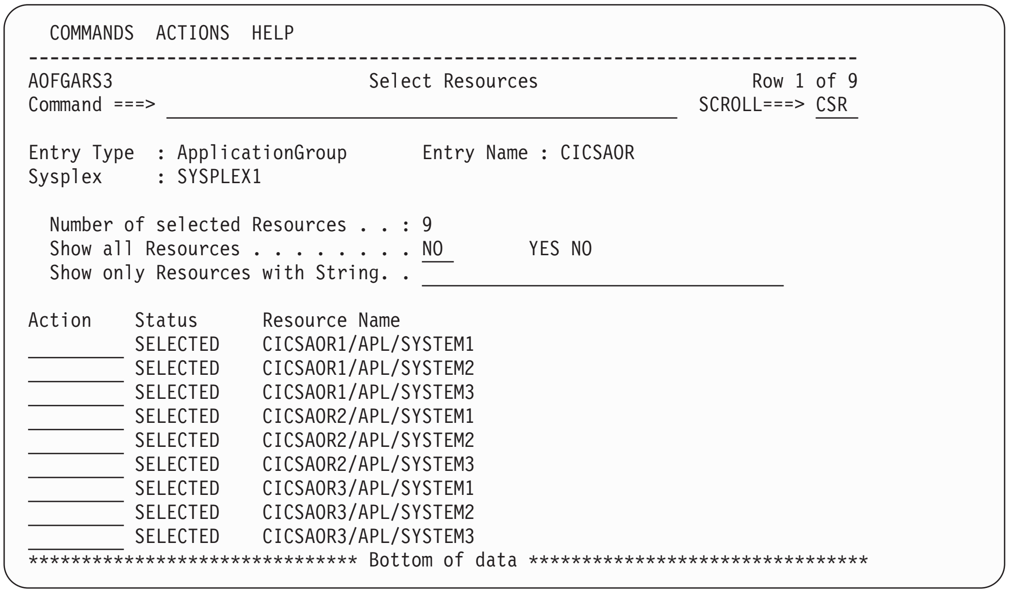


Figure 28. Current Configuration with Applications CICSAOR1 through CICSAOR3

11. On the Policy Selection panel for CICSAOR select the RESOURCES policy item. The Select Resources panel is displayed with 9 application resources selected. You can see this panel in Figure 29 on page 55.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS3                        Select Resources                    Row 1 of 9
Command ===> ________________________________________________  SCROLL===> CSR_

Entry Type  : ApplicationGroup       Entry Name : CICSAOR
Sysplex     : SYSPLEX1

  Number of selected Resources . . : 9
  Show all Resources . . . . . . . . NO_       YES NO
  Show only Resources with String. . __________________________________

Action    Status     Resource Name
_________ SELECTED    CICSAOR1/APL/SYSTEM1
_________ SELECTED    CICSAOR1/APL/SYSTEM2
_________ SELECTED    CICSAOR1/APL/SYSTEM3
_________ SELECTED    CICSAOR2/APL/SYSTEM1
_________ SELECTED    CICSAOR2/APL/SYSTEM2
_________ SELECTED    CICSAOR2/APL/SYSTEM3
_________ SELECTED    CICSAOR3/APL/SYSTEM1
_________ SELECTED    CICSAOR3/APL/SYSTEM2
_________ SELECTED    CICSAOR3/APL/SYSTEM3
****************************** Bottom of data ********************************
```

Figure 29. Resources for the CICSAOR Sysplex Application Group

12. On the Select Resources panel for CICSAOR, deselect the three resources on SYSTEM1, because CICS should not run on SYSTEM1. Type an `M` (the action code for `REMOVE`) in the **Action** column in front of each resource that you want to remove from the group.

    When you press Enter, the **Number of selected Resources** field now shows a `6`.
13. Press PF3 (END) to return to the Policy Selection panel for CICSAOR.
14. Select policy item APPLGROUP INFO to get to the Application Group Information panel for CICSAOR.
15. Change the **Automatically link Application-Resources** field to `NO`.

    Reason: Application group definitions are complete now. Further modifications within the sysplex should not affect this completed resource selections on panel Select Resources.

> **Background:**
> *Automatically link Application Resources*
>
> Set this input field to NO if the definition of an application group resource (with its linked applications, its nature and its behavior and so on) is complete. This prevents possible changes in the automation policy, like changed system links or application links, from having an influence on this application group resource.
>
> If this field is set to YES, the application group resource will be changed if system links or application links occur, namely, the number of generated application resources, the number of linked application resources and possibly the behavior of the application group resource will change.
>
> For example, if you link another system to SYSPLEX1, you would get three more resources into your application group CICSAOR, one for each linked application.
>
> Later on, you will specify preference values for application resources, where automatically generated resources with their default preferences could adversely affect your complete application group definition.

16. Press PF3 until you are back on the Entry Type Selection panel.

    Now application group CICSAOR is complete: if a start request is issued against CICSAOR, also start requests to CICSAOR1, CICSAOR2 and CICSAOR3 are automatically issued on SYSTEM2 and SYSTEM3.

Step 3.2: Create System Application Group DB2

The application group DB2 runs on two specific systems, so you define it as a System Application Group.

1. On the Entry Type Selection panel, enter option 5 or APG.

   The Entry Name Selection panel is displayed with the application group CICSAOR.
2. Enter `NEW DB2`

   The Define New Entry panel for application groups is displayed; the **Name** field is already filled with `DB2`.
3. In the scenario you decided to let DB2 run permanently only on two of the three systems of SYSPLEX1: SYSTEM2 and SYSTEM3. Thus you type `SYSTEM` in the **Application Group Type** field.

   You merely want DB2 be started with all of its components but without specific backup considerations. Thus you leave the **Nature** field as `BASIC`.

   For resource generation, enter `DB2` in the **Automation Name** field.
4. Press PF3 to reach the Policy Selection panel.

   > **Note:**
   > A resource is not created for the APG at this point in time, because the application group is not yet linked to one or more systems.

   The Policy Selection panel is displayed indicating that the application group has been created.

5. Select the WHERE USED policy item. The Where Used panel is displayed with the three systems.
6. DB2 should only run on SYSTEM2 and SYSTEM3. Select SYSTEM2 and SYSTEM3 and press PF3 (END). You will see the Command Progress Display popup window telling you that the following two resources have been created:

   ```
   DB2/APG/SYSTEM2
   DB2/APG/SYSTEM3
   ```

Background:

Generating System Application Group Resources

A System Application Group resource is automatically generated if there is:

- An Automation Name for the application group, and
- A link to a system.

for each linked system, one resource is generated.

There are two ways to link a System Application Group to a system:

- For the application group use the WHERE USED policy item on the Policy Selection panel or the WHEREUSED (W) command on the Entry Name Selection panel to get to the list of systems, and then select at least one system.
- If you are on the Policy Selection panel for a system, select the APPLICATION GROUPS policy item and then select an application group.

The current configuration is shown in Figure 30.

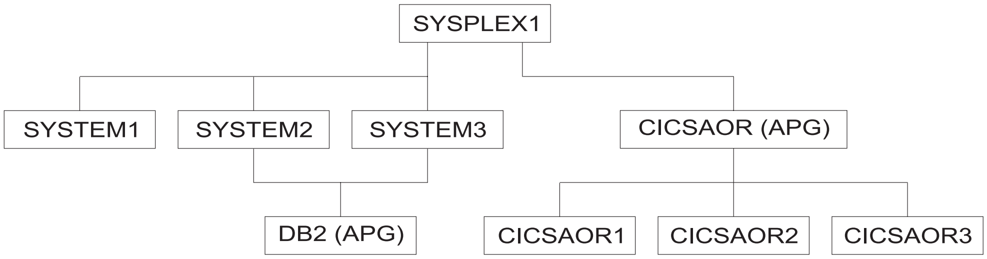


Figure 30. Current Configuration with Application Group DB2

7. On the Policy Selection panel for DB2, select the APPLICATIONS policy item to link the required applications.

   The Applications for ApplicationGroup panel is displayed with all twelve applications.
8. Select DB2MAIN and DB2IRLM and press PF3 (END) to return to the Policy Selection panel.

   Now DB2MAIN and DB2IRLM are linked to two systems via the application group DB2. For the application group, the field *Automatically link Application-Resources into APG* is set to YES, which is the default.

One resource is generated for each system link of each application. The Command Progress Display popup window indicates that the following four resources are generated:

```
DB2IRLM/APL/SYSTEM2
DB2IRLM/APL/SYSTEM3
DB2MAIN/APL/SYSTEM2
DB2MAIN/APL/SYSTEM3
```

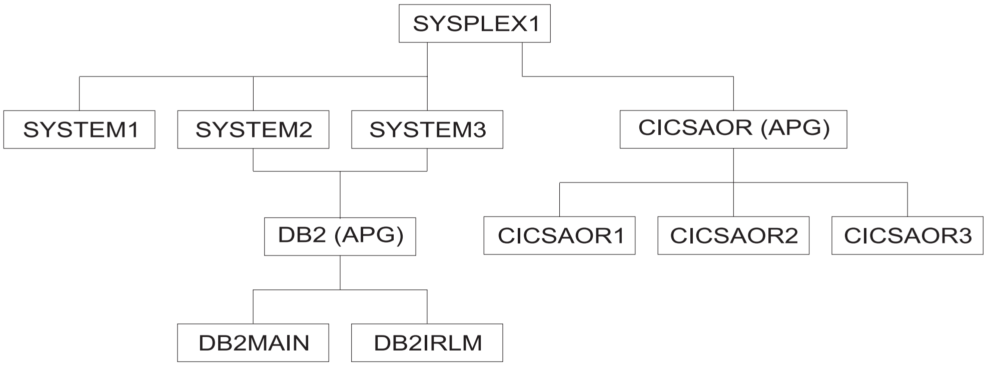
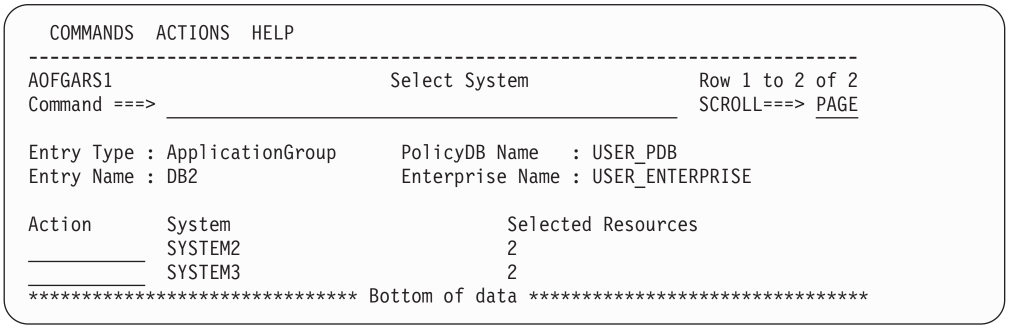


Figure 31. Current Configuration with Applications Linked to DB2

9. On the Policy Selection panel for application group DB2, select the RESOURCES policy item.

   The Select System panel with an indication of the number of resources selected for each system is shown. You can compare the resulting panel on your system with Figure 32.



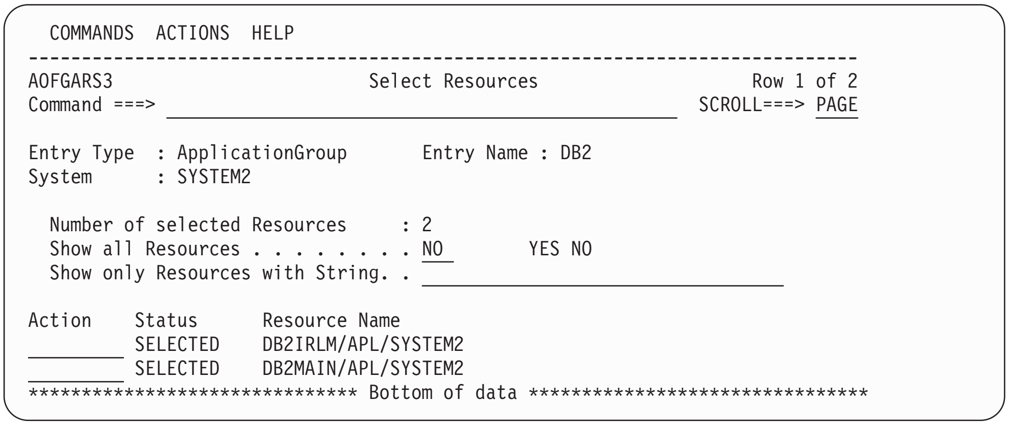
```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS1                          Select System              Row 1 to 2 of 2
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : DB2                   Enterprise Name : USER_ENTERPRISE

Action       System                          Selected Resources
___________  SYSTEM2                         2
___________  SYSTEM3                         2
******************************* Bottom of data ********************************
```

Figure 32. Select System Panel for DB2 Application group

A System Application Group can be linked to several systems, which generates application resources for those systems. So to work with the resource links, you have to specify first which system you want to get the resources for. This is done on the Select System panel by selecting one of the systems.

10. On the Select System panel, select either of the SYS Resources SYSTEM2 or SYSTEM3 and press Enter.

    The Select Resources panel with, for example, the two APL Resources for SYSTEM2 is displayed. You can compare the resulting panel on your system with Figure 33 on page 59.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS3                        Select Resources                   Row 1 of 2
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type  : ApplicationGroup       Entry Name : DB2
System      : SYSTEM2

  Number of selected Resources     : 2
  Show all Resources . . . . . . . . NO_       YES NO
  Show only Resources with String. . __________________________________

Action    Status      Resource Name
_________ SELECTED    DB2IRLM/APL/SYSTEM2
_________ SELECTED    DB2MAIN/APL/SYSTEM2
****************************** Bottom of data *******************************
```

Figure 33. Select Resources Panel for DB2 Application Group

Both applications should be started, thus no change on this panel is required.

11. On the Select Resources panel, press PF3 (END) or CANCEL. This returns you to the Select System panel, from which you can view the Select Resources panel for the APL resources in SYSTEM3.
12. Press PF3 (END) twice on the Select Resources panel with APL Resources for SYSTEM3 to return to the Policy Selection panel for application group DB2.
13. Select policy item APPLGROUP INFO to reach the Application Group Information panel. Change the **Automatically link Application-Resources into APG** field to `NO`.

    The DB2 application group is now complete.
14. Press PF3 until you are back at the Entry Type Selection panel.
15. The application group DB2 is now complete: if a start request is issued against DB2 on SYSTEM2, start requests to the application resources DB2MAIN and DB2IRLM are also automatically issued for SYSTEM2. A start request against application group DB2 on SYSTEM3 does the same for SYSTEM3.

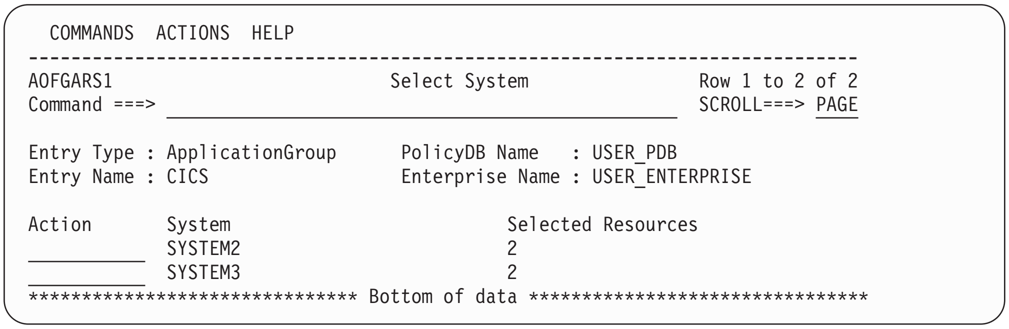
Step 3.3: Create System Application Group CICS

You will create this application group with the nature of BASIC and with links to three applications (CICSMAIN, CICSTOR, CICSFOR). These applications are already defined.

1. On the Entry Type Selection panel, enter option 5 or APG. To create the CICS System Application Group you have to repeat steps 2 through 10 of “Step 3.2: Create System Application Group DB2” on page 56 accordingly.

   On the Applications for ApplicationGroup panel, select applications

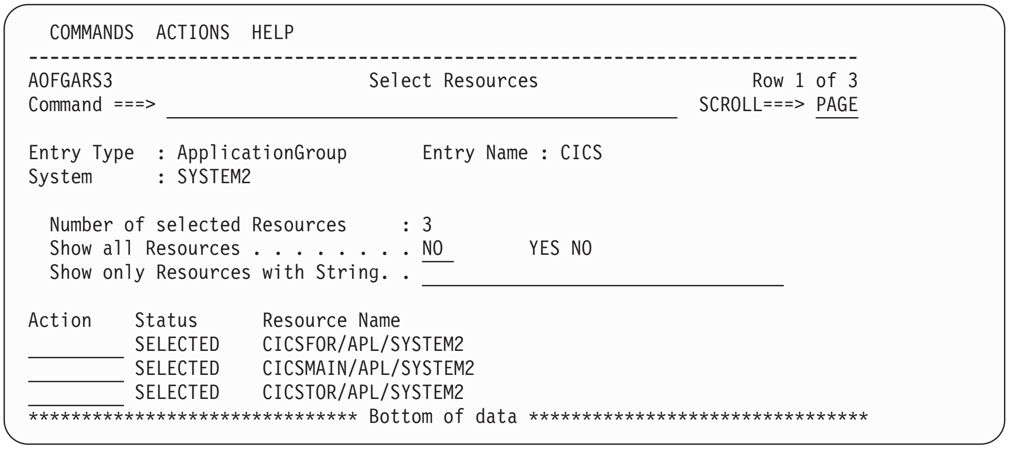
   ```
   CICSMAIN
   CICSTOR
   CICSFOR
   ```
2. The Select System panel is displayed as shown in Figure 34 on page 60.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS1                          Select System              Row 1 to 2 of 2
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : CICS                  Enterprise Name : USER_ENTERPRISE

Action       System                          Selected Resources
___________  SYSTEM2                         2
___________  SYSTEM3                         2
******************************* Bottom of data ********************************
```

Figure 34. Grouping Resources

Select SYSTEM2. A panel is displayed similar to the one shown in Figure 35.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS3                        Select Resources                  Row 1 of 3
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type  : ApplicationGroup       Entry Name : CICS
System      : SYSTEM2

  Number of selected Resources     : 3
  Show all Resources . . . . . . . . NO_       YES NO
  Show only Resources with String. . __________________________________

Action    Status      Resource Name
_________ SELECTED    CICSFOR/APL/SYSTEM2
_________ SELECTED    CICSMAIN/APL/SYSTEM2
_________ SELECTED    CICSTOR/APL/SYSTEM2
******************************* Bottom of data ********************************
```

Figure 35. Resources for the CICS Sysplex Application Group

Check the created links for SYSTEM3 too. The selected links should be the same.

3. Press PF3 to get back to the Policy Selection panel and select policy item APPLGROUP INFO. Once again, change the **Automatically link Application-Resources** field to NO.
4. Press PF3 until you are back on the Entry Type Selection panel.

   Now application group CICS is complete: if a start request is issued against CICS, start requests are also issued to application resources CICSFOR, CICSMAIN, CICSTOR (on SYSTEM2 and SYSTEM3).

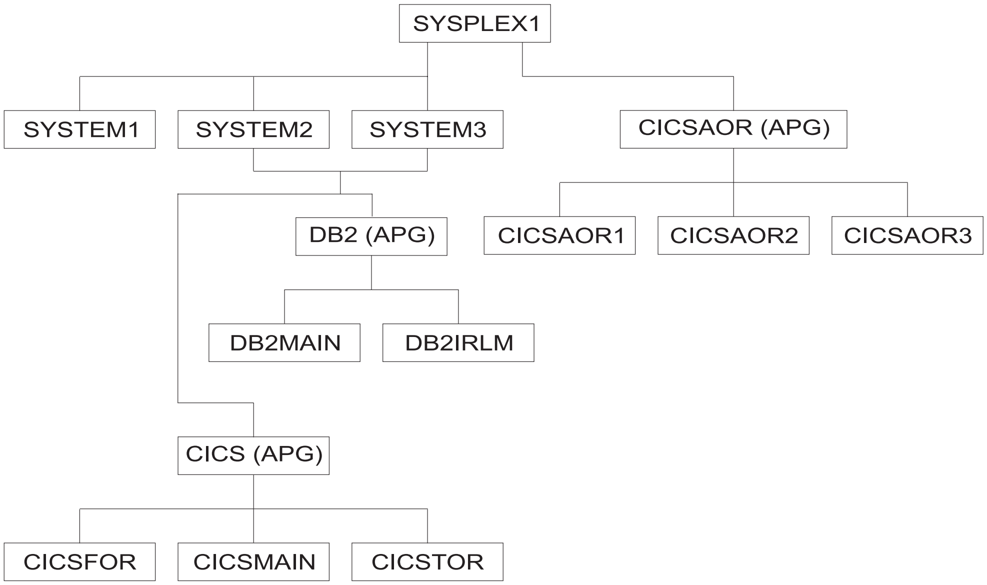


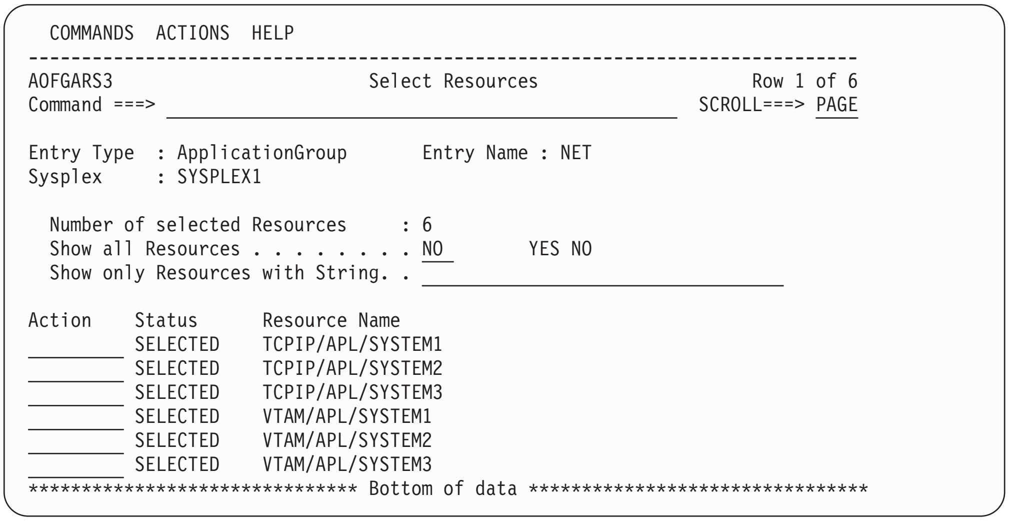
Figure 36. Current Configuration with Applications Linked to DB2

Step 3.4: Create Sysplex Application Group NET

The creation of Sysplex Application Group NET follows the previous examples, therefore, no detailed description of the required steps is contained here.

The Sysplex Application Group NET should run in the entire SYSPLEX1, and the applications VTAM and TCP/IP should be linked to it.

After your definitions are complete, the Select Resources panel for the NET application group should look like Figure 37.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS3                        Select Resources                    Row 1 of 6
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type  : ApplicationGroup       Entry Name : NET
Sysplex     : SYSPLEX1

  Number of selected Resources     : 6
  Show all Resources . . . . . . . . NO        YES NO
  Show only Resources with String. . __________________________________

Action    Status      Resource Name
_________ SELECTED    TCPIP/APL/SYSTEM1
_________ SELECTED    TCPIP/APL/SYSTEM2
_________ SELECTED    TCPIP/APL/SYSTEM3
_________ SELECTED    VTAM/APL/SYSTEM1
_________ SELECTED    VTAM/APL/SYSTEM2
_________ SELECTED    VTAM/APL/SYSTEM3
******************************* Bottom of data ********************************
```

Figure 37. Select Resources Panel forApplication Group NET

Step 3.5: Create MOVE Group DWHAPG

The Data Warehouse Group DWHAPG should run on one system (SYSTEM2) and, if something goes wrong on SYSTEM2, alternatively on SYSTEM3. This application group is created with nature MOVE.

Background:
What are Move Groups?

All previous application groups were of nature BASIC: BASIC means that all of its components are peers and all of them need to be available before the application group is considered to be available.

A MOVE Group, as a contrast, is designed for backup solutions. A MOVE Group can have many components, but exactly one is supposed to be active. All other components are for backup only. Thus MOVE Groups usually contain members of one kind (where one application can replace the others).

The applications needed for the DWHAPG application group are available. Thus no additional preparation seem to be needed (an error, as you will see).

1. On the Entry Type Selection panel, enter option `5` or `APG`. The Entry Name Selection panel is displayed with four application groups: CICS, CICSAOR, DB2, NET.
2. Enter `NEW DWHAPG` at the command line. The Define New Entry panel is displayed; the **Name** field is already filled with `DWHAPG`.
3. The application group DWHAPG should run alternatively on two out of the three systems. Thus leave the **Application Group Type** field as `SYSPLEX`.

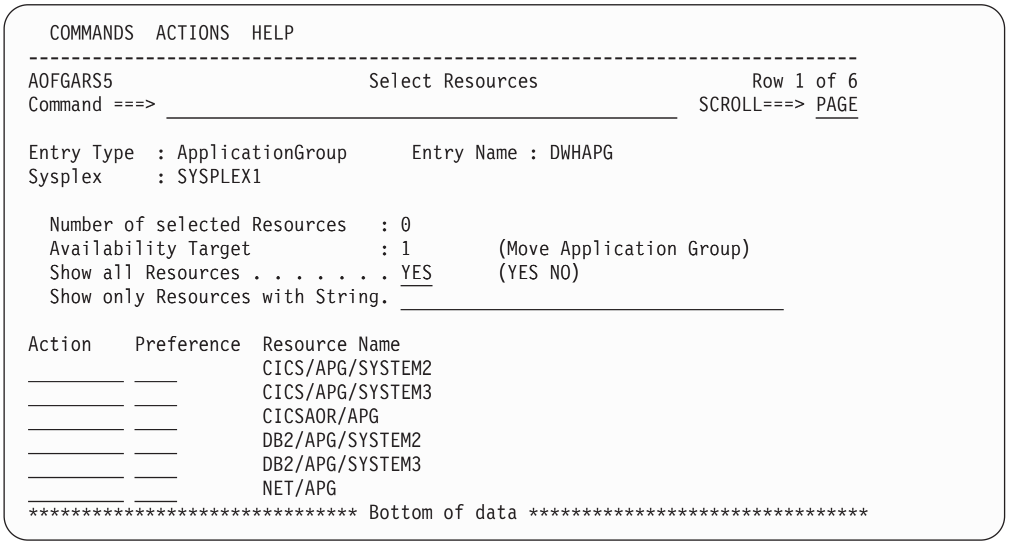
   You want to start the components of DWHAPG exactly once, but with backups. Thus you change its nature to `MOVE`.

   For resource generation you need an automation name. Therefore enter `DWHAPG` in the **Automation Name** field.
4. Press PF3 to exit the panel. The Policy Selection panel for Groups is displayed showing a message that the application group has been created.
5. On the Policy Selection panel select the WHERE USED policy item. The Where Used panel is displayed with sysplex group SYSPLEX1.
6. Select the sysplex group and press PF3.

   You will see the Command Progress Display panel informing you that the `DWHAPG/APG` resource has been created.
7. Select the RESOURCES policy item. You will see the Select Resources panel with an empty list of resources. The reason why no resource seems to be available is that you did not link any applications to the application group - thus no application resources were generated.

   Note that the Availability Target is 1 (exactly 1 resource is supposed to be up in a MOVE Group). You cannot change this value.
8. Change the **Show all Resources** field to `YES`

   The Select Resources panel is displayed with all valid application group resources. Valid application group resources are Sysplex Application Group resources on the same sysplex (CICSAOR/APG, NET/APG) and System Application Group resources on one of the systems in the Sysplex (CICS/APG/SYSTEM2, CICS/APG/SYSTEM3, DB2/APG/SYSTEM2 and DB2/APG/SYSTEM3). Figure 38 on page 63 shows this panel.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS5                        Select Resources                   Row 1 of 6
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type  : ApplicationGroup      Entry Name : DWHAPG
Sysplex     : SYSPLEX1

  Number of selected Resources   : 0
  Availability Target            : 1       (Move Application Group)
  Show all Resources . . . . . . . YES     (YES NO)
  Show only Resources with String. ____________________________________

Action    Preference  Resource Name
_________ ____        CICS/APG/SYSTEM2
_________ ____        CICS/APG/SYSTEM3
_________ ____        CICSAOR/APG
_________ ____        DB2/APG/SYSTEM2
_________ ____        DB2/APG/SYSTEM3
_________ ____        NET/APG
******************************** Bottom of data ********************************
```

Figure 38. Select Resources Panel for DWHAPG with valid APG Resources

However, the resulting panel shown in Figure 38 is not yet what you wanted: The target of your group was to start all of these components plus the DWH Application on either SYSTEM2 or SYSTEM3.

The target, to start the whole package alternatively on SYSTEM2 or SYSTEM3 cannot be reached with the described definitions.

Solution: You have to define two additional application groups with the same contents - one with focus on SYSTEM2 and one with focus on SYSTEM3. Their names will be DWHAPG2 and DWHAPG3. DWHAPG will have these two APGs as member and start them alternatively.

However, you must define these groups as System Application Groups because they must have the System Application Group CICS on SYSTEM2 and SYSTEM3.

9. Press PF3 until you are back on panel Entry Name Selection and enter `NEW DWHAPG2` at the command line.

   Or, as a fast path, specify `=APG;NEW DWHAPG2`.

   The Define New Entry panel is displayed, with DWHAPG2 already filled in the **Name** field.

10. DWHAPG2 should run on a single system. Therefore, select type SYSTEM on the Define New Entry panel.

    As you want to start all of DWHAPG2's components, select BASIC.

    For resource generation you need an automation name. Type `DWHAPG2` in the related field and press PF3 to exit the panel.

11. On the Policy Selection panel select the WHERE USED policy item and on the Where Used panel select sysplex group SYSTEM2.

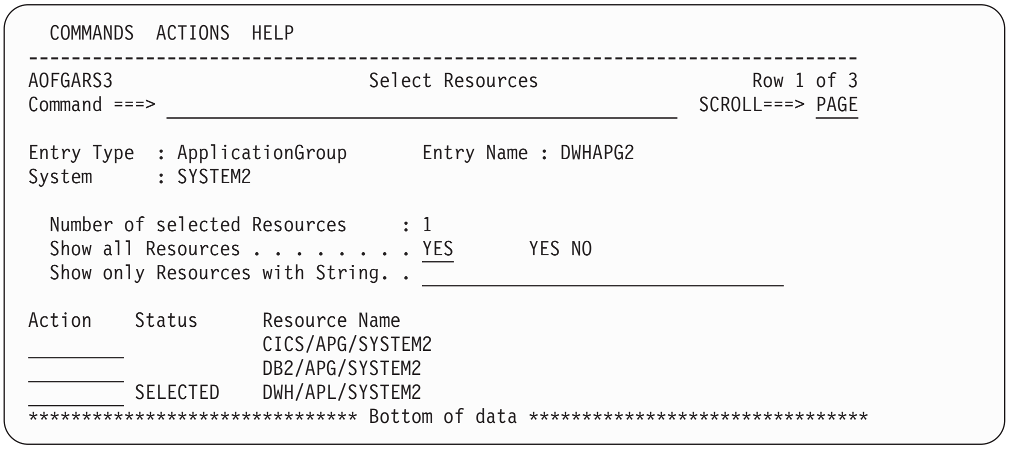
12. Press PF3 to exit the panel. The Command Progress Display panel is displayed, showing that the resource DWHAPG2/APG/SYSTEM2 has been generated.

13. Select policy item APPLICATIONS. From the displayed list, select application DWH. Press PF3 to exit the panel.

    As a result, the application resource DWH/APL/SYSTEM2 is generated and linked to the application group resource DWHAPG2.

14. Select the RESOURCES policy item. The Select Resources panel with the APL Resource DWH/APL/SYSTEM2 is displayed.

15. Select this APL Resource, if it is not already selected.
16. Change the **Show all Resources** field to YES. As a result, two application group resources are displayed together with the one application resource, as shown in Figure 39.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS3                        Select Resources                    Row 1 of 3
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type  : ApplicationGroup       Entry Name : DWHAPG2
System      : SYSTEM2

  Number of selected Resources     : 1
  Show all Resources . . . . . . . . YES       YES NO
  Show only Resources with String. . __________________________________

Action    Status      Resource Name
                      CICS/APG/SYSTEM2
_________             DB2/APG/SYSTEM2
_________             
_________ SELECTED    DWH/APL/SYSTEM2
******************************* Bottom of data ********************************
```

Figure 39. Select Resources Panel for DWHAPG2

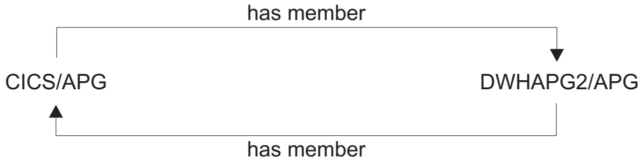
17. You want to use DWHAPG2 to start the application groups CICS and DB2 on SYSTEM2.

    Select CICS on SYSTEM2 (that is, resource CICS/APG/SYSTEM2), DB2 on SYSTEM2 (that is, resource DB2/APG/SYSTEM2) and DWH on SYSTEM2 (see Figure 39).

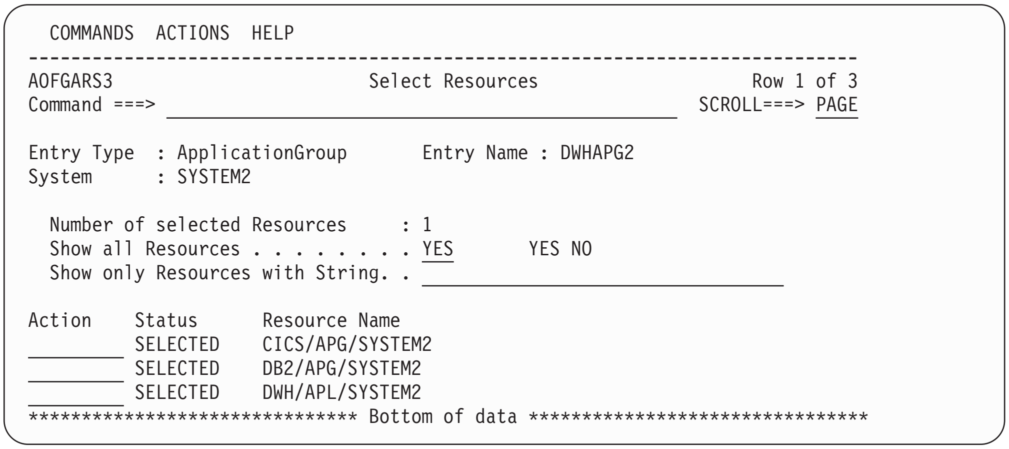
Background:

Defining Loops in Groups

The application group CICS is now linked to DWHAPG2. However, when you are using the resource selection panel for CICS, be careful not to link DWHAPG2 to CICS, because this would result in a loop.



18. Change the **Show selected Resources only** field to YES to check the result. The Select Resources panel with one application resource and two application group resources selected is displayed, as shown in Figure 40 on page 65).



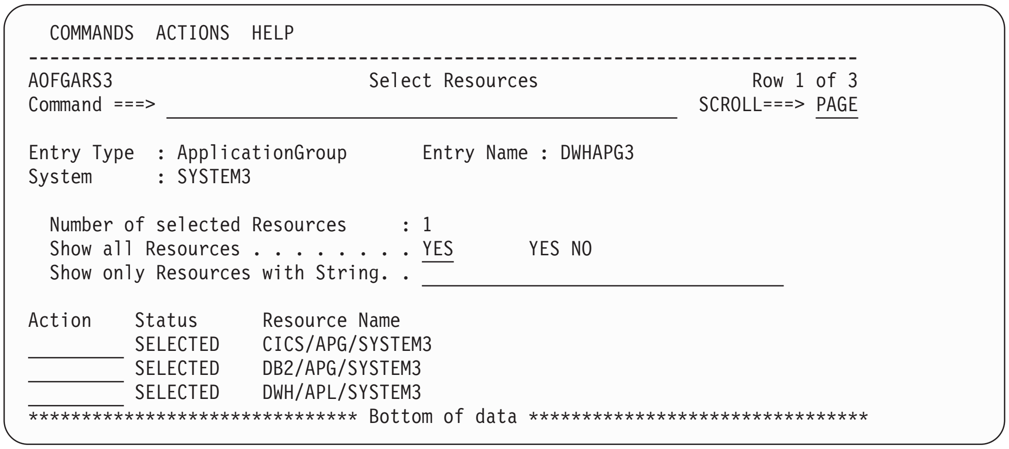
```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS3                        Select Resources                    Row 1 of 3
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type  : ApplicationGroup       Entry Name : DWHAPG2
System      : SYSTEM2

  Number of selected Resources     : 1
  Show all Resources . . . . . . . . YES       YES NO
  Show only Resources with String. . __________________________________

Action    Status      Resource Name
_________ SELECTED    CICS/APG/SYSTEM2
_________ SELECTED    DB2/APG/SYSTEM2
_________ SELECTED    DWH/APL/SYSTEM2
****************************** Bottom of data ********************************
```

Figure 40. Select Resources Panel for DWHAPG2

19. Press PF3 until you return to the Entry Name Selection panel.
20. Now create application group DWHAPG3 in the same way as application group DWHAPG2, as described in steps 9 on page 63 to 12 on page 63, and link it to System SYSTEM3.
21. For application group DWHAPG3, you want the same application members as you already defined for DWHAPG2, but with focus on SYSTEM3. For this purpose, you select the COPY policy item and in the subsequent Select Entry for Copy panel, you select DWHAPG2.

    You should be aware that COPY only copies parts of source-objects data and only parts of the links. For more information, see “Common Policy Items” on page 41.

    After copying and pressing PF3, you are now back on the Policy Selection panel for DWHAPG3.
22. Select the RESOURCES policy item and link the resources for SYSTEM3. The result should look like the panel shown in Figure 41.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS3                        Select Resources                    Row 1 of 3
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type  : ApplicationGroup       Entry Name : DWHAPG3
System      : SYSTEM3

  Number of selected Resources     : 1
  Show all Resources . . . . . . . . YES       YES NO
  Show only Resources with String. . __________________________________

Action    Status      Resource Name
_________ SELECTED    CICS/APG/SYSTEM3
_________ SELECTED    DB2/APG/SYSTEM3
_________ SELECTED    DWH/APL/SYSTEM3
****************************** Bottom of data ********************************
```

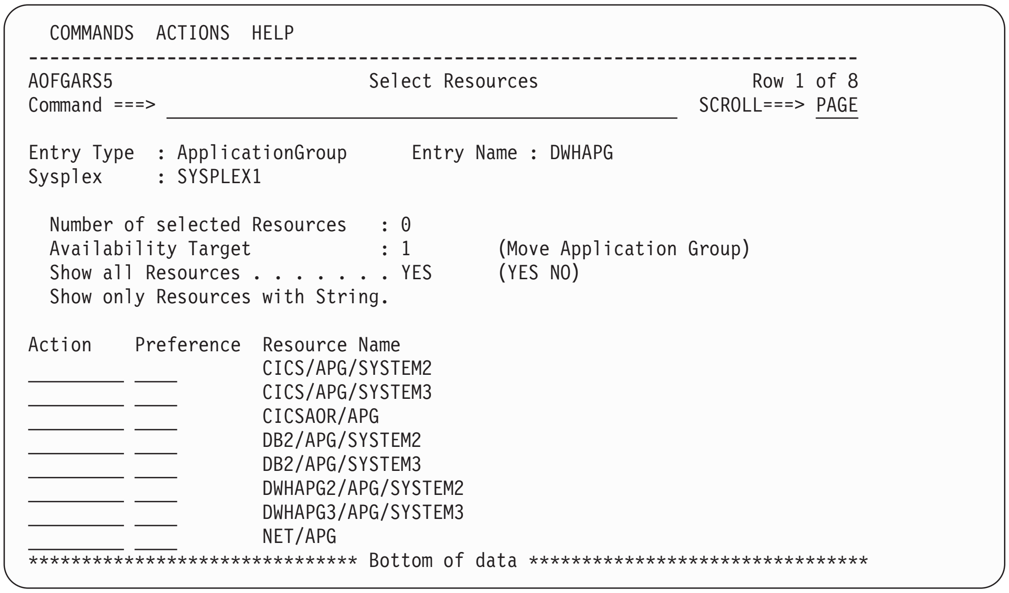
Figure 41. Select Resources Panel for DWHAPG3

    Press PF3 until you are back on the Entry Name Selection panel.

23. Now you have all components for your Move Group DWHAPG, and you can select DWHAPG on the Entry Name Selection panel. Select its RESOURCES policy item.
24. The Select Resources panel for DWHAPG is displayed.

    Change the **Show all Resources** field to `YES`.

The Select Resources panel with eight APG resources is displayed (see Figure 42).



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS5                        Select Resources                    Row 1 of 8
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type  : ApplicationGroup      Entry Name : DWHAPG
Sysplex     : SYSPLEX1

  Number of selected Resources   : 0
  Availability Target            : 1        (Move Application Group)
  Show all Resources . . . . . . . YES      (YES NO)
  Show only Resources with String.

Action    Preference  Resource Name
_________ ____        CICS/APG/SYSTEM2
_________ ____        CICS/APG/SYSTEM3
_________ ____        CICSAOR/APG
_________ ____        DB2/APG/SYSTEM2
_________ ____        DB2/APG/SYSTEM3
_________ ____        DWHAPG2/APG/SYSTEM2
_________ ____        DWHAPG3/APG/SYSTEM3
_________ ____        NET/APG
******************************* Bottom of data ********************************
```

Figure 42. Select Resources Panel for DWHAPG with Valid APG Resources

Background:
Using Preferences to Define Specific Roles

The Resources panel for MOVE groups looks slightly different than the panels for BASIC groups. It has an additional field (**Availability Target**) that cannot be modified, and the second column in the scrollable part of the panel is called **Preferences**.

Preferences are used to define specific roles of the resources linked to a group. You want DWHAPG2 on SYSTEM2 be started first and, if this fails, you want to start DWHAPG3 on SYSTEM3. To define this, start with preference values 700 and 500. They have the following meaning:

700 Resource is started when the group is started. The resource is not stopped unless a very high preference (greater than 950) alternative becomes available. (Use for the resource that normally runs).

500 Resource is started when the higher preference alternative becomes unavailable. It continues running when normal Preference (700) alternatives become available again. It stops at next (scheduled) outage. (Use for the backup resource).

For more details about preference values, refer to “RESOURCES Policy Item” on page 148.

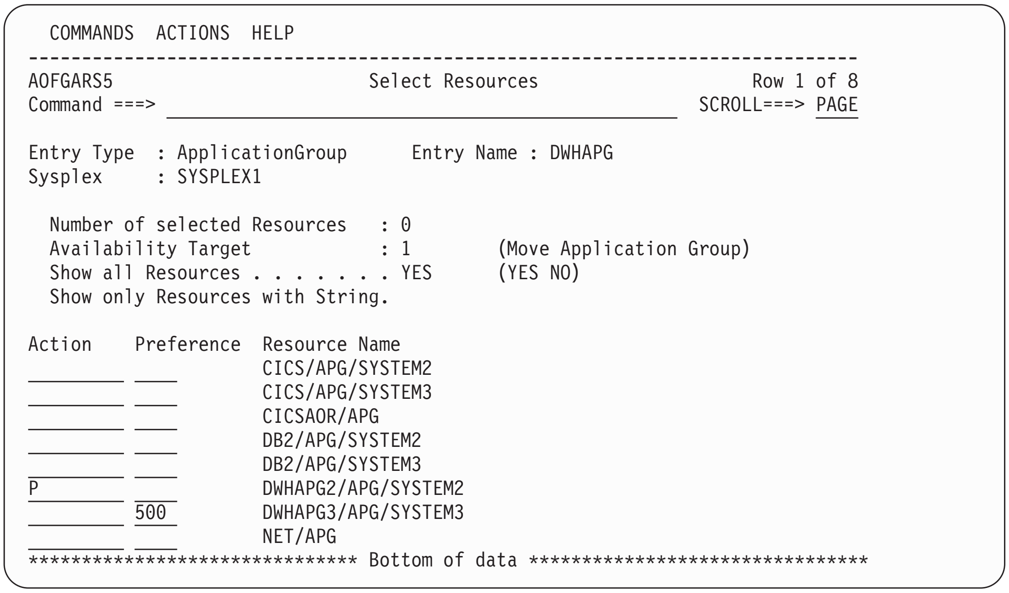
The SECON(D)ARY command sets a preference of 400, which has the following meaning:

400 Resource is started when the higher preference alternative becomes unavailable. The difference to preference 500 is that the resource is stopped when the normal Preference (700) alternative becomes available again.

25. Set the preferences: Select the DWHAPG2 group on SYSTEM2 as primary (using action P in the **Action** column) and the DWHAPG3 group on SYSTEM3 as secondary (using preference value 500 in the **Preference** column). Figure 43 on page 68 shows the Select Resources panel with the selections made and Figure 44 on page 68 shows the Select Resources panel after you pressed Enter: with the action P converted to the corresponding numeric value 700.

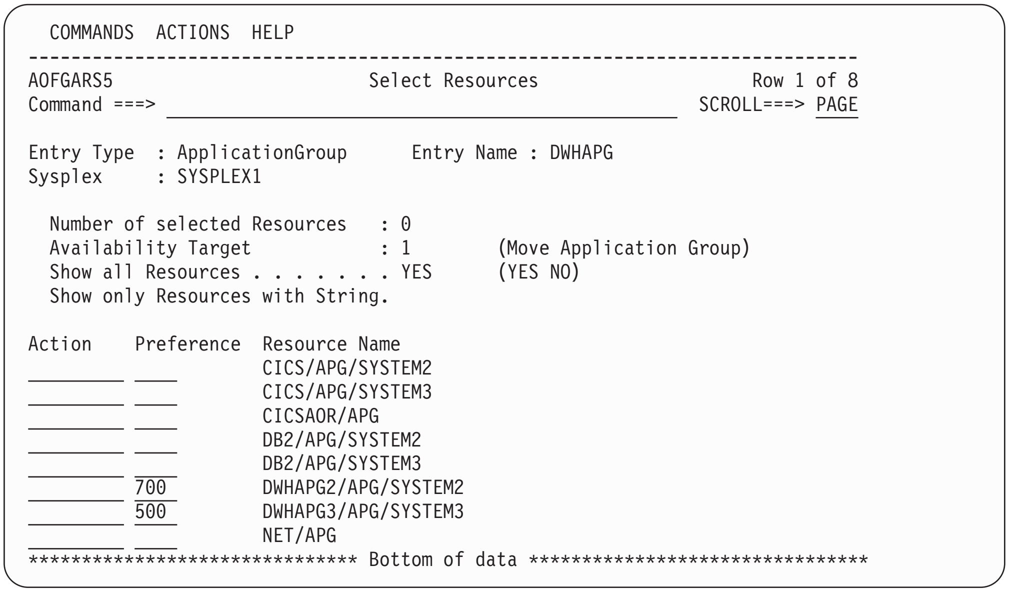
Note:
You can either set preferences directly in the **Preference** column or use the **Action** column for important values:

- PRIMARY...(P): Sets ″primary″ preference of 700.
- SECONDARY.(D): Sets ″secondary″ preference of 400.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS5                        Select Resources                   Row 1 of 8
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type  : ApplicationGroup      Entry Name : DWHAPG
Sysplex     : SYSPLEX1

  Number of selected Resources   : 0
  Availability Target            : 1       (Move Application Group)
  Show all Resources . . . . . . . YES     (YES NO)
  Show only Resources with String.

Action    Preference  Resource Name
_________ ____        CICS/APG/SYSTEM2
_________ ____        CICS/APG/SYSTEM3
_________ ____        CICSAOR/APG
_________ ____        DB2/APG/SYSTEM2
_________ ____        DB2/APG/SYSTEM3
P________ ____        DWHAPG2/APG/SYSTEM2
_________ 500_        DWHAPG3/APG/SYSTEM3
_________ ____        NET/APG
******************************* Bottom of data ********************************
```

Figure 43. Select Resources Panel for DWHAPG with Selected Preferences



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS5                        Select Resources                   Row 1 of 8
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type  : ApplicationGroup      Entry Name : DWHAPG
Sysplex     : SYSPLEX1

  Number of selected Resources   : 0
  Availability Target            : 1       (Move Application Group)
  Show all Resources . . . . . . . YES     (YES NO)
  Show only Resources with String.

Action    Preference  Resource Name
_________ ____        CICS/APG/SYSTEM2
_________ ____        CICS/APG/SYSTEM3
_________ ____        CICSAOR/APG
_________ ____        DB2/APG/SYSTEM2
_________ ____        DB2/APG/SYSTEM3
_________ 700_        DWHAPG2/APG/SYSTEM2
_________ 500_        DWHAPG3/APG/SYSTEM3
_________ ____        NET/APG
******************************* Bottom of data ********************************
```

Figure 44. Select Resources Panel for DWHAPG with Converted Preferences

Press PF3 to return to the Policy Selection panel for DWHAPG.

26. Select policy item APPLGROUP INFO and change the **Automatically link Application-Resources** field to NO.
27. Now press PF3 until you are back on the Entry Type Selection panel.

    Application group DWHAPG is complete now: if a start request is issued against DWHAPG, its primary resource DWHAPG2 is started. If this fails, a start vote is issued against DWHAPG3.

Step 3.6: Create Sysplex Application Group WBSRVAPG

In this step you create the Sysplex Application Group WBSRVAPG as a SERVER group.

Background:

What is a SERVER Group?

SERVER groups are created when not all but only a certain number of its linked resources need to be available in order to consider the application group as being available.

Thus a SERVER group with an *Availability Target* of 1 and the preference values set accordingly can behave like a MOVE group.

On the other hand, a SERVER group with Availability Target set to *ALL and all the Preferences of its members set to 700, will act like a BASIC group.

The Web server group WBSRVAPG should run only on SYSTEM1. The WBSRV application and the NET application group are essential parts that must be up. In contrast, the DB2 application group is not necessarily needed.

1. On the Entry Type Selection panel, enter `5` or `APG`.

   The Entry Name Selection panel is displayed and shows seven application groups.
2. Enter `NEW WBSRVAPG` at the command line.

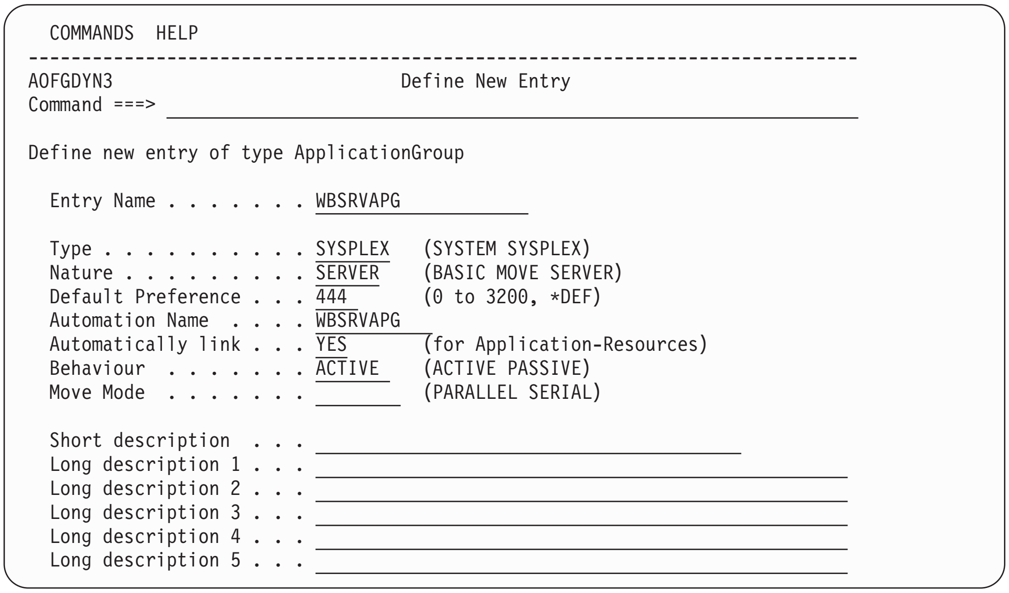
   The Define New Entry panel is displayed; the **Name** field is already filled with `WBSRVAPG`.
3. The application group WBSRVAPG runs mainly on SYSTEM1. But WBSRVAPG will have some members that run on other systems from the sysplex. Thus you leave the **Application Group Type** field as `SYSPLEX`.

   WBSRVAPG should start with all of its components, but it is acceptable if DB2 cannot be started. Thus you change the nature of the WBSRVAPG group to `SERVER`.

   On the same panel, in the **Default Preference** field , change the default preference to `444` as the initial value for all resources that will be linked automatically.

   For resource generation you need an automation name; so enter `WBSRVAPG` in the **Automation Name** field.

   The input in the Define New Entry panel should now look similar to Figure 45 on page 70.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN3                           Define New Entry
Command ===> _________________________________________________________________

Define new entry of type ApplicationGroup

  Entry Name . . . . . . . WBSRVAPG

  Type . . . . . . . . . . SYSPLEX   (SYSTEM SYSPLEX)
  Nature . . . . . . . . . SERVER    (BASIC MOVE SERVER)
  Default Preference . . . 444       (0 to 3200, *DEF)
  Automation Name  . . . . WBSRVAPG
  Automatically link . . . YES       (for Application-Resources)
  Behaviour  . . . . . . . ACTIVE    (ACTIVE PASSIVE)
  Move Mode  . . . . . . . ________  (PARALLEL SERIAL)

  Short description  . . . ________________________________________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 45. Define New Entry Input Panel for WBSRVAPG

4. Press PF3 to create WBSRVAPG. The Policy Selection panel for WBSRVAPG is displayed with a message indicating that the application group has been created.
5. On the Policy Selection panel select the WHERE USED policy item. The Where Used panel is displayed with sysplex group SYSPLEX1.
6. Select the sysplex group and press PF3. You will see the Command Progress Display popup window telling you that the resource WBSRVAPG/APG has been created.
7. Select policy item APPLICATIONS of WBSRVAPG and link application WBSRV. Press PF3 to return to the Policy Selection panel for WBSRVAPG. You will be notified that the following resources have been generated:

   ```
   WBSRV/APL/SYSTEM1
   WBSRV/APL/SYSTEM2
   WBSRV/APL/SYSTEM3
   ```

8. Select the RESOURCES policy item of WBSRVAPG to get to the Select Resources panel for WBSRVAPG.

   As you can see, the Preference for the linked resources is the value you entered on the Define New Entry panel for application groups.

   You can use the **Preference** column to fine-tune the automatic linking of application resources into an application group.

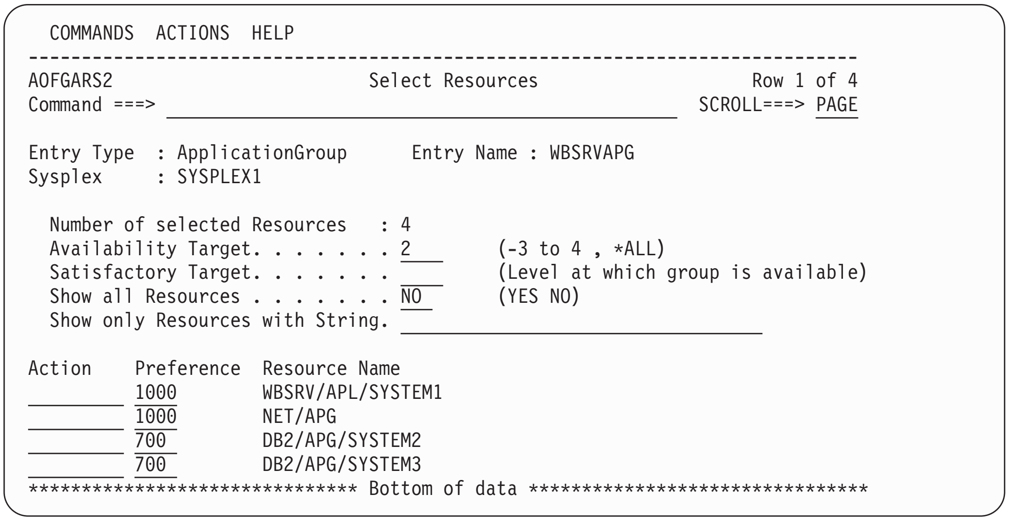
   By setting the **Show all Resources** field to `YES`, you get the complete list of (twelve) selectable resources.

   Change the preferences so that the WBSRVAPG application group is started on SYSTEM1 with WBSRV and NET as mandatory components and DB2 as not-mandatory. To achieve this:

   - Set the **Availability Target** to `2`
   - Select mandatory components (WBSRV/APL/SYSTEM1 and NET/APG) with a preference value of 1000.
   - Deselect WBSRV/APL/SYSTEM2 and WBSRV/APL/SYSTEM3 (action `m`).
   - Select DB2 (DB2/APG/SYSTEM2 and DB2/APG/SYSTEM3) with a preference value between 501 and 999.

When the application group is started, all selected members with at least Preference 600 will also be started, but members with Preference 1000 are mandatory.

If you change the **Show all Resources** field to NO, your definitions should look like the sample screen shown in Figure 46.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS2                        Select Resources                   Row 1 of 4
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type  : ApplicationGroup      Entry Name : WBSRVAPG
Sysplex     : SYSPLEX1

  Number of selected Resources   : 4
  Availability Target. . . . . . . 2___      (-3 to 4 , *ALL)
  Satisfactory Target. . . . . . . ____      (Level at which group is available)
  Show all Resources . . . . . . . NO_       (YES NO)
  Show only Resources with String. __________________________________

Action    Preference  Resource Name
_________ 1000        WBSRV/APL/SYSTEM1
_________ 1000        NET/APG
_________ 700         DB2/APG/SYSTEM2
_________ 700         DB2/APG/SYSTEM3
****************************** Bottom of data ********************************
```

Figure 46. Final Select Resources Panel for WBSRVAPG

9. The WBSRVAPG application group is now complete. Therefore, press PF3 in the Select Resources panel, select policy item APPLGROUP INFO in the Policy Selection panel and once again change the **Automatically link Application-Resources into APG** field to NO on the Application Group Information panel.

   Now the application group WBSRVAPG is complete: If a start request is issued against WBSRVAPG, at least WBSRV and NET must be successfully started to make WBSRVAPG available.

How to Use Schedules (Service Periods)

You can extend the scenario described in “A Grouping Scenario” on page 12 using service periods that automatically control the startup and shutdown of resources. Service periods can be linked to APLs, APGs, and MTRs.

You will extend the scenario so that:

- The data warehouse application is available from 9 a.m. to 10 p.m.
- The Web server is available from 4 a.m. through 11 p.m.

Three service periods are needed:

- SWBSRV linked to the WBSRVAPG (Web server) application group
- SDWH linked to the DWHAPG (data warehouse) application group
- SDB2DWN linked to the DB2 application group

The SDB2DWN service period should be for the resource to be down 24 hours a day, seven days a week. This will serve to shut down DB2 and CICS when they are not needed to support an application (WBSRV or DWH or anything else that is dependent upon them).

Schedule Override

The Web server needs to be available until midnight. So the operator just needs to override the SWBSRV service period (changing 11 p.m. to 12 a.m.). An appropriate action would be the only one required to reverse the override.

If you do not want to use service periods, you could just issue a high priority request for WBSRVAPG to be available sometime before 11 p.m.. To remove this request at midnight, you could either use INGREQ command with the expiration time parameter, which will automatically remove the request. Or, at 12 a.m., you could use the INGVOTE command to remove this request.

Early Start

At 3 a.m. the Web server application needs to be started.

The operator simply creates an override for SWBSRV that says to start the Web server at 3 a.m. and keep it up until 11 p.m.. Everything will then happen automatically. This can be done several hours (or even days) in advance.

If the operator does not want to use service periods to achieve this, he or she just needs to issue a start command against the WBSRVAPG application group and all the required resources will be made available (and kept available) until the operator removes their request (if this is after 4 a.m., the resources will not be stopped because the desired status of the resource will not change. The requested AVAILABLE from the operator will be replaced with a requested AVAILABLE from the schedule).

How to Use Triggers and Events

This section presents a scenario on how to define automatic starts and stops based either on fixed time intervals (service periods) or on special conditions (events).

The scenario consists of a task and a solution that gives:

- A description of what policy objects and links you need and an explanation of why you need them (“The Solution” on page 73)
- Step-by-step instructions on how to enter these in the customization dialog (“Implementing the Solution” on page 76)

Note that, like service periods, triggers can be linked to APLs, APGs, and MTRs

The Task

Assume that you are in the environment that you created in “How to Work with Resources” on page 47 and that you want to set up the following CICS applications:

- One terminal-owning region (TOR) on system SYSTEM3
- One application-owning region (AOR) on SYSPLEX1 to support this TOR

They should meet these requirements:

1. Between 7 a.m. and 6 p.m. on weekdays and between 8 a.m. and 2 p.m. on Saturdays:
   a. The TOR should be available.
   b. The AOR should be available on one of the systems in SYSPLEX1.

2. Before any of the CICS applications are started, certain related data sets need to be backed up.
3. A TOR does not make sense without an AOR, therefore:
   a. The TOR should not be started until the AOR is available on one of the systems.
   b. Whenever the AOR has to be shut down altogether for some reason (such as an emergency) so that it is no longer available on *any* of the systems in the sysplex, the TOR needs to be shut down *first*.

Note: It does not matter which system the AOR is available on and it need not be the same system throughout the specified periods. Thus it is enough to start the AOR on one of the systems and switch to another system only when the AOR cannot or does not continue to run on its current system.

The Solution

Figure 47 on page 74 shows the policy objects and connections that you need to define in the customization dialog.

Figure 48 on page 74 shows the resources that will be generated from your policy objects.

Note: Some of the policy objects that you need here were already created in “How to Work with Resources” on page 47. So you can start from that existing policy database. If you do, you will have more entries and links in your policy database than are present in Figure 47 on page 74, but the additional ones will not affect the solution of the current task.

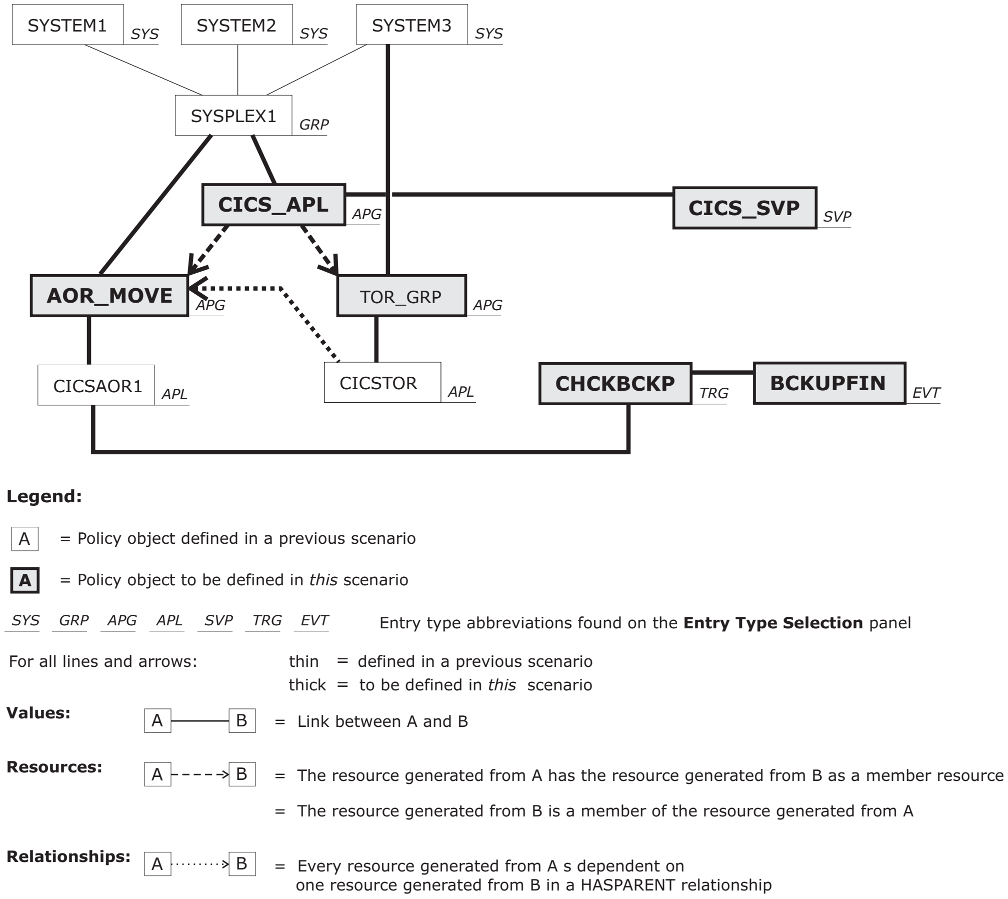

SYSTEM1 SYS
SYSTEM2 SYS
SYSTEM3 SYS
SYSPLEX1 GRP
CICS_APL APG
CICS_SVP SVP
AOR_MOVE APG
TOR_GRP APG
CICSAOR1 APL
CICSTOR APL
CHCKBCKP TRG
BCKUPFIN EVT
Legend:
A = Policy object defined in a previous scenario
A = Policy object to be defined in this scenario
SYS GRP APG APL SVP TRG EVT Entry type abbreviations found on the Entry Type Selection panel
For all lines and arrows: thin = defined in a previous scenario
thick = to be defined in this scenario
Values: A B = Link between A and B
Resources: A B = The resource generated from A has the resource generated from B as a member resource
= The resource generated from B is a member of the resource generated from A
Relationships: A B = Every resource generated from A s dependent on one resource generated from B in a HASPARENT relationship

Figure 47. Policy Objects in the Solution

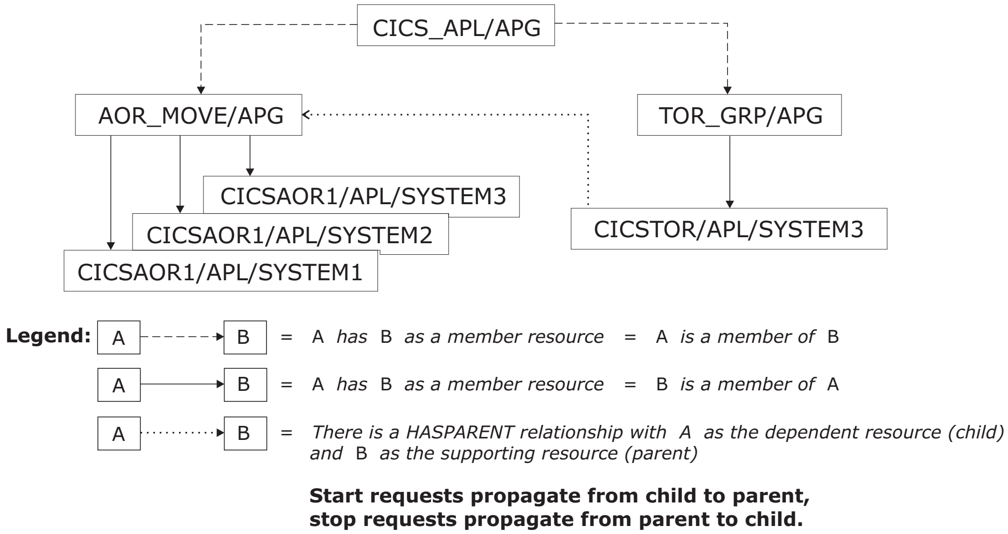

CICS_APL/APG
AOR_MOVE/APG
TOR_GRP/APG
CICSAOR1/APL/SYSTEM3
CICSAOR1/APL/SYSTEM2
CICSAOR1/APL/SYSTEM1
CICSTOR/APL/SYSTEM3
Legend: A B = A has B as a member resource = A is a member of B
A B = A has B as a member resource = B is a member of A
A B = There is a HASPARENT relationship with A as the dependent resource (child) and B as the supporting resource (parent)
Start requests propagate from child to parent, stop requests propagate from parent to child.

Figure 48. Generated Resources in the Solution

The AOR can run on different systems in the sysplex. This means you create *one* policy object of type APL to represent the AOR, but SA z/OS generates *three* resources from this policy object (one resource per system in the sysplex).

Background:

SA z/OS makes a distinction between the following:

- Applications (that is, policy objects of type APL)
- Application *resources*

What you define in the customization dialog are applications, not application resources. Application resources are *generated* from the applications that you define. Application resources are not policy objects.

For your convenience, you can use application CICSAOR1 as the AOR. (You defined CICSAOR1 in "How to Work with Resources" on page 47.)

The TOR needs to be generated from a different APL policy object, of course. You can use application CICSTOR for this purpose, which you defined in "How to Work with Resources" on page 47.

Resources will only be generated from a policy object if it is linked (directly or indirectly) to a system or sysplex. For entry type APL, the customization dialog does *not* allow you to create a *direct* link. What you need to do is link both the application and the system or sysplex to the same application *group*. In the plan shown in Figure 47 on page 74, this is done as follows:

- Application group AOR_MOVE provides indirect linking between application CICSAOR1 and sysplex SYSPLEX1
- Application group TOR_GRP provides indirect linking between application CICSTOR and system SYSTEM3.

To deal with requirement 1 on page 72 (availability times), you need to define an appropriate service period (that is, a policy object of type SVP). This is CICS_SVP in Figure 47 on page 74. Details are given in step 17 on page 80. There are two *alternative* ways to meet requirement 1:

1. Link the service period to both application group AOR_MOVE and application group TOR_GRP.
2. Create an additional policy object that *groups* these two application groups and then link the service period to that additional policy object.

The plan in Figure 47 on page 74 uses alternative 2. The additional policy object that groups AOR_MOVE and TOR_GRP is application group CICS_APL.

To deal with requirement 2 on page 73 (backing up data sets), you need to define an appropriate event (that is: a policy object of type EVT) and combine it with a trigger (that is: a policy object of type TRG). The event is BCKUPFIN in Figure 47 and the trigger is CHCKBCKP. Details are given in steps 13, 14, 15, and 16 in "Implementing the Solution" on page 76.

To deal with requirement 3 on page 73 (starting and stopping resources in the correct order), you need to define an appropriate HASPARENT relationship (see Figure 9 on page 16 for an explanation of HASPARENT). In defining the relationship, you need to choose a supporting resource. The application resources generated so far would be a bad choice because they would not produce the desired effect. Therefore, the solution is to define an additional move group (that is, policy object of type APG and nature MOVE), which is AOR_MOVE in

Figure 47 on page 74. The move group *resource* that SA z/OS *generates* from AOR_MOVE can then be used as the supporting resource in the HASPARENT relationship.

Implementing the Solution

Step 1. On the Customization Dialog Primary Menu panel, open the policy database that you created in “How to Work with Resources” on page 47.

Step 2. Define a policy object of type APG named `AOR_MOVE` with these values:

- **Application Group Type**: SYSPLEX
- **Nature**: MOVE
- **Automation Name**: AOR_MOVE

Prerequisite for the next steps:
In “How to Work with Resources” on page 47, you defined SYSPLEX1 as a policy object of type GRP and linked it to SYSTEM1, SYSTEM2 and SYSTEM3.

Step 3. Link application group AOR_MOVE to group SYSPLEX1.

Result: SA z/OS generates the APG resource: `AOR_MOVE/APG`

Prerequisite for the next steps:
In “How to Work with Resources” on page 47, you defined CICSAOR1 as a policy object of type APL.

Step 4. Link application CICSAOR1 to application group AOR_MOVE.

Result: SA z/OS generates the following APL resources:

```
CICSAOR1/APL/SYSTEM1
CICSAOR1/APL/SYSTEM2
CICSAOR1/APL/SYSTEM3
```

Step 5. In application group AOR_MOVE, select policy item RESOURCES and set the following preferences for the new APL resources:

- 900 for CICSAOR1/APL/SYSTEM1
- 600 for CICSAOR1/APL/SYSTEM2
- 500 for CICSAOR1/APL/SYSTEM3

Meaning of the preferences:

- Starting AOR_MOVE/APG will amount to starting CICSAOR1/APL/SYSTEM1.
- CICSAOR1/APL/SYSTEM2 will only be started if CICSAOR1/APL/SYSTEM1 fails.
- CICSAOR1/APL/SYSTEM3 will only be started if CICSAOR1/APL/SYSTEM2 fails.

Step 6. Define a policy object of type APG named `TOR_GRP` with these values:

- **Application Group Type**: SYSTEM
- **Nature**: BASIC
- **Automation Name**: TOR_GRP

Prerequisite for the next steps:
In “How to Work with Resources” on page 47, you defined SYSTEM3 as a policy object of type SYS.

Step 7. Link application group TOR_GRP to system SYSTEM3.

Result: SA z/OS generates the APG resource: `TOR_GRP/APG/SYSTEM3`

Prerequisite for the next steps:
In “How to Work with Resources” on page 47, you defined CICSTOR as a policy object of type APL.

Step 8. Link application CICSTOR to application group TOR_GRP.

Result: SA z/OS generates the APL resource: `CICSTOR/APL/SYSTEM3`

Step 9. Define a HASPARENT relationship between APL resource CICSTOR/APL/SYSTEM3 as the child (dependent resource) and APG resource AOR_MOVE/APG as the parent (supporting resource) as follows:

Step a. In application CICSTOR, select policy item RELATIONSHIPS.

Step b. On the Relationship Selection List panel that comes up, enter `new hasparent` at the command line.

Step c. On the Define Relationship panel that comes up, enter resource `AOR_MOVE/APG` in the **Supporting Resource** field.

Note: The resource name needs to be entered by typing it in. There is no list of resources to select from.

Step 10. Define a policy object of type APG named `CICS_APL` with these values:

- **Application Group Type**: SYSPLEX
- **Nature**: BASIC
- **Automation Name**: CICS_APL

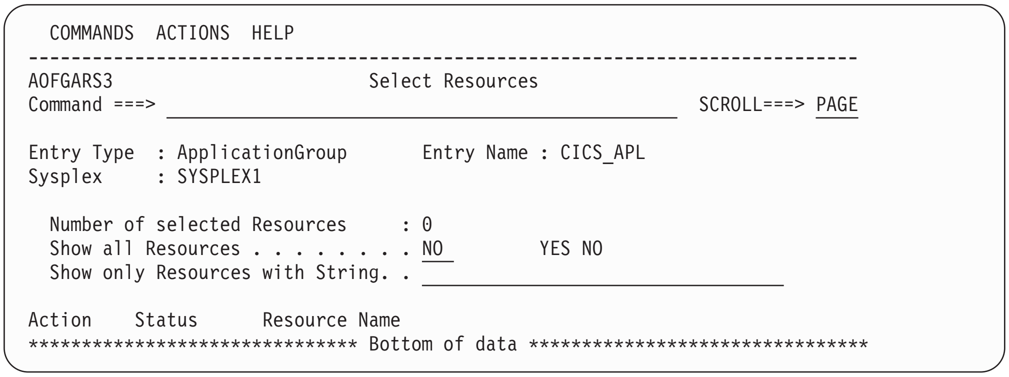
Step 11. Link application group CICS_APL to group SYSPLEX1.

Result: SA z/OS generates the APG resource: `CICS_APL/APG`

Step 12. Make APG resources TOR_GRP/APG/SYSTEM3 and AOR_MOVE/APG members of APG resource CICS_APL/APG as follows:

Step a. In application group CICS_APL, select policy item RESOURCES.

This displays the Select Resources panel with `NO` in the **Show all Resources** field, as shown in Figure 49 on page 78.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS3                        Select Resources
Command ===>                                                  SCROLL===> PAGE

Entry Type  : ApplicationGroup       Entry Name : CICS_APL
Sysplex     : SYSPLEX1

  Number of selected Resources     : 0
  Show all Resources . . . . . . . . NO          YES NO
  Show only Resources with String. .

Action    Status      Resource Name
******************************* Bottom of data ********************************
```

Figure 49. Select Resources Panel

Step b. Overtype the NO with YES and press Enter.

This will display the resources that have been generated so far (except for CICS_APL/APG itself). They will be listed in the **Resource Name** column.

Step c. Use action s to change the status of the following resources to SELECTED:

- AOR_MOVE/APG
- TOR_GRP/APG/SYSTEM3

This makes these resources members of CICS_APL/APG.

Note: You can verify the result by selecting policy item MEMBER OF in application group TOR_GRP and in application group AOR_MOVE.

Step 13. Define a policy object of type EVT (an event) named BCKUPFIN.

Purpose: This event is meant to be *set* whenever the backup process that is mentioned in the task has completed (see requirement 2 on page 73). The administrator can have SA z/OS automatically set this event by trapping message *IEF404I* from the backup process. Use the INGEVENT command to set events.

Step 14. In event BCKUPFIN, select policy item CONDITION and, on the Unset Condition panel that appears, enter the option that suits your needs in the **Unset Condition** field.

Background:

The condition for unsetting an event will be checked on the *observed* status. There are eleven possible values for the observed status: six that indicate availability and five that indicate unavailability (see *IBM Tivoli System Automation for z/OS User's Guide* for details). Among those values that indicate availability, there is one named STARTING. The "UP" condition, for example, means: the observed status has a value that indicates availability, but is *not* the value STARTING.

The observed status belongs to generated resources, not to policy objects. In other words: a policy object does not have an observed status. Only the resources that are *generated from* the policy object have an observed status. So there is an observed status for each of the boxes in Figure 48 on page 74, but not for the boxes in Figure 47 on page 74.

As a result of all this, the condition for unsetting an event only has an effect when the event is related to a generated resource because only then is there an observed status that the condition can be checked on.

The only way that an event can be related to generated resources is via a *trigger*, as follows: when the event is linked to a trigger and the trigger is linked to an APL or APG policy object, the event will be *indirectly* related to every resource that is generated from that APL or APG policy object.

The key to understanding the effect of the unset condition is that SA z/OS maintains a *separate* event value for each generated resource that an event is indirectly related to. So an event may be set for one generated resource, but unset for another at the same time.

The effect of UP, for example, is that the event will be unset *for a particular generated resource* as soon as the observed status of *that* particular generated resource is given a value that indicates availability, but is different from the value STARTING.

Step 15. Define a policy object of type TRG (a trigger) named `CHCKBCKP`, then enter trigger details as follows (to make the trigger check backup completion before startups):

Step a. On the Policy Selection panel for the new trigger, select policy item CONDITION.

Step b. On the Trigger Conditions panel that comes up, select condition STARTUP 1.

Step c. On the STARTUP Condition for Trigger panel that follows, use action `s` to change the status of event BCKUPFIN to SELECTED.

Step 16. Link trigger CHCKBCKP to application CICSAOR1 by proceeding as follows:

Step a. In trigger CHCKBCKP, select policy item WHERE USED.

Step b. On the Where Used panel that comes up, use action s to change the status of application CICSAOR1 to SELECTED.

Result: As a result of linking trigger CHCKBCKP to both application CICSAOR1 (16 on page 79) and event BCKUPFIN (15 on page 79), SA z/OS will maintain a separate BCKUPFIN value for each of the three resources that were generated from application CICSAOR1 in 4 on page 76:

- One BCKUPFIN value for CICSAOR1/APL/SYSTEM1
- One BCKUPFIN value for CICSAOR1/APL/SYSTEM2
- One BCKUPFIN value for CICSAOR1/APL/SYSTEM3

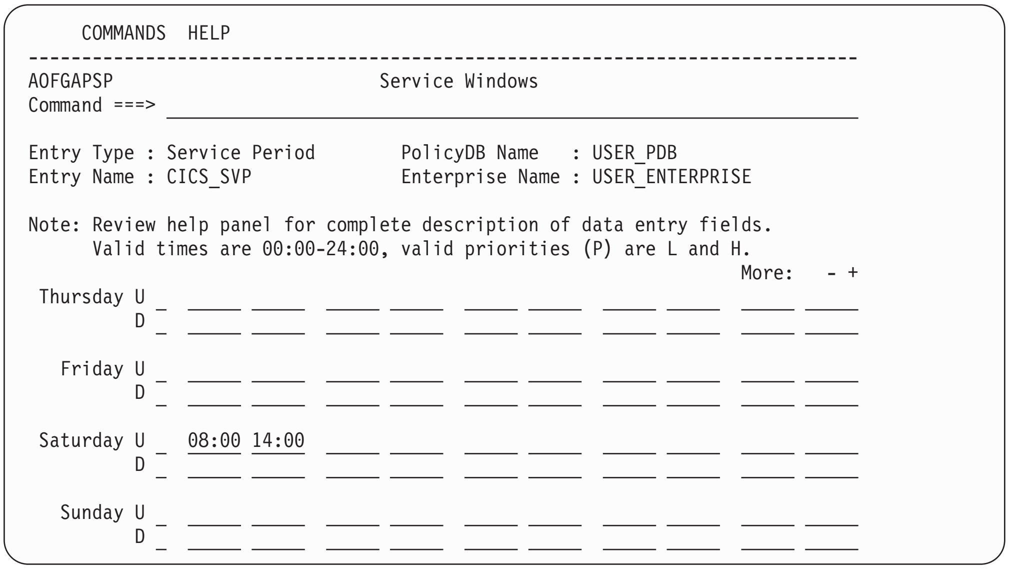
When you use INGEVENT to set the event, you may supply the name of one of these resources (as described in *IBM Tivoli System Automation for z/OS Operator's Commands*) so that the event is only set for that particular resource, but remains unset for the others. If you do not supply a resource parameter with INGEVENT, the event will be set for all three resources in this scenario. They will, however, still be *independent* of each other, so they might be unset at different times.

Step 17. Define a policy object of type SVP (a service period) named `CICS_SVP`, then enter schedule details as follows:

Step a. On the Policy Selection panel for the new service period, select policy item SERVICE WINDOW.

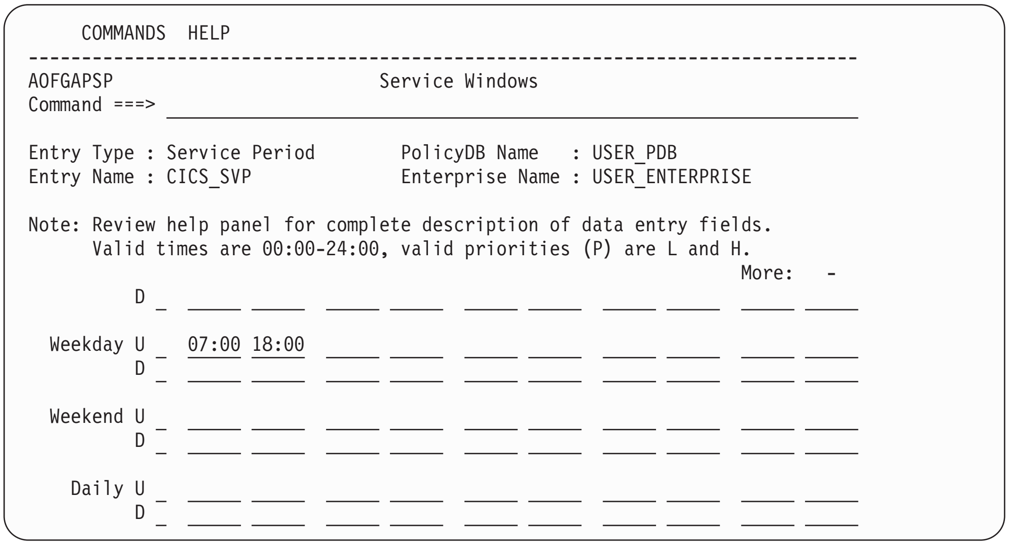
Step b. On the Service Windows panel that comes up, scroll down until you see the lines for SATURDAY and (further down) for WEEKDAY and fill in the values shown in Figure 50 and Figure 51 on page 81.

Note: For further information on the Service Windows panel see "Service Periods Policy Object" on page 235.



```
     COMMANDS  HELP
------------------------------------------------------------------------------
AOFGAPSP                           Service Windows
Command ===> _________________________________________________________________

Entry Type : Service Period          PolicyDB Name   : USER_PDB
Entry Name : CICS_SVP                Enterprise Name : USER_ENTERPRISE

Note: Review help panel for complete description of data entry fields.
      Valid times are 00:00-24:00, valid priorities (P) are L and H.
                                                                 More:   - +
 Thursday U _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

   Friday U _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

 Saturday U _  08:00 14:00  _____ _____  _____ _____  _____ _____  _____ _____
          D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

   Sunday U _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
```

Figure 50. Saturday Details on the Service Windows Panel



```
     COMMANDS  HELP
------------------------------------------------------------------------------
AOFGAPSP                         Service Windows
Command ===> _________________________________________________________________

Entry Type : Service Period       PolicyDB Name   : USER_PDB
Entry Name : CICS_SVP             Enterprise Name : USER_ENTERPRISE

Note: Review help panel for complete description of data entry fields.
      Valid times are 00:00-24:00, valid priorities (P) are L and H.
                                                                  More:   -
         D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

  Weekday U _  07:00 18:00  _____ _____  _____ _____  _____ _____  _____ _____
          D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

  Weekend U _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

    Daily U _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
```

Figure 51. Weekday Details on the Service Windows Panel

Step 18. Link service period CICS_SVP to application group CICS_APL as follows:

Step a. In service period CICS_SVP, select policy item WHERE USED.

Step b. On the Where Used panel that comes up, use action s to change the status of application group CICS_APL to SELECTED.

How the Solution Works

The key to understanding how this works is the distinction in SA z/OS between *starting* a resource and setting a resource's *desired status* to AVAILABLE (see *IBM Tivoli System Automation for z/OS User's Guide* for more information on the concept of desired status).

A desired status of AVAILABLE is a prerequisite for starting a resource, but it may not be enough. Further startup prerequisites may be attached to individual resources, for example, conditions that are implicit in a HASPARENT relationship.

In addition, in order to understand how this works, you need to be aware of what it actually means to start a resource. For an APL resource, this is obvious: it means that a start order is sent to the appropriate automation agent. For an APG resource, however, it means something different, namely that the start request is propagated to some or all (depending on the APG resource's nature) of its *member* resources. Provided that the propagated start request *wins* over any other request that might be simultaneously sent to the same member resource for independent causes, the member resource's desired status is set to AVAILABLE.

The resources in our scenario and the special prerequisites attached to them can be seen in Figure 48 on page 74. This is what happens:

1. Whenever the time of day enters the time window defined for availability in service period CICS_SVP, SA z/OS will generate a start request to resource CICS_APL/APG.
2. If this start request wins over any concurrent requests to CICS_APL/APG from elsewhere, SA z/OS will set the desired status of CICS_APL/APG to AVAILABLE.
3. Resource CICS_APL/APG has no further start prerequisites attached to it, so it is started immediately.

4. The type of CICS_APL/APG is APG and its nature is BASIC, so to start CICS_APL/APG means that the start request is propagated to *every* resource that is a member of it. In this scenario, there are two such resources:
   - AOR_MOVE/APG
   - TOR_GRP/APG
5. If the start request that is propagated to AOR_MOVE/APG wins over any concurrent requests from elsewhere, SA z/OS will set the desired status of AOR_MOVE/APG to AVAILABLE, and if the start request that is propagated to TOR_GRP/APG wins, SA z/OS will set the desired status of TOR_GRP/APG to AVAILABLE.
6. Neither AOR_MOVE/APG nor TOR_GRP/APG has any further start prerequisites attached to it, so when their desired status is set to AVAILABLE, they are started immediately.
7. To start TOR_GRP/APG means that the start request is propagated to its only member, which is CICSTOR/APL/SYSTEM3.
8. If the start request that is propagated to CICSTOR/APL/SYSTEM3 wins over any concurrent requests from elsewhere, SA z/OS will set the desired status of CICSTOR/APL/SYSTEM3 to AVAILABLE.
9. However, because of the HASPARENT relationship that is attached to CICSTOR/APL/SYSTEM3, a desired status of AVAILABLE is not yet enough to start CICSTOR/APL/SYSTEM3. So nothing happens as long as its supporting resource, AOR_MOVE/APG, has not started up.
10. AOR_MOVE/APG, on the other hand, is a *move* group, so to start AOR_MOVE/APG means that the start request is only propagated to *one* of its member resources, namely the one with the highest preference (referred to as its *primary member*). Because of step 5 on page 76, this member is CICSAOR1/APL/SYSTEM1.
11. If the start request that is propagated to CICSAOR1/APL/SYSTEM1 wins over any concurrent requests from elsewhere, SA z/OS will set the desired status of CICSAOR1/APL/SYSTEM1 to AVAILABLE.
12. Because of the trigger, however, a desired status of AVAILABLE is not yet enough to start CICSAOR1/APL/SYSTEM1. So nothing happens *before* the backup event is set for resource CICSAOR1/APL/SYSTEM1. As soon as the event is set for resource CICSAOR1/APL/SYSTEM1, CICSAOR1/APL/SYSTEM1 will start up.
13. CICSAOR1/APL/SYSTEM1 is the first resource to start up that is of type APL rather than APG. So to start CICSAOR1/APL/SYSTEM1 means that the automation manager will send a start order for CICSAOR1/APL/SYSTEM1 to the automation agent on SYSTEM1.
14. When CICSAOR1/APL/SYSTEM1 is fully up and running, the automation agent on SYSTEM1 will report the fact back to the automation manager, which will set the *observed* status of CICSAOR1/APL/SYSTEM1 to an observed status value that indicates availability, but is different from STARTING (see *IBM Tivoli System Automation for z/OS User's Guide* for details on observed status values).
15. The new observed status passes up from CICSAOR1/APL/SYSTEM1 to AOR_MOVE/APG because AOR_MOVE/APG is a move group and CICSAOR1/APL/SYSTEM1 is its primary member. So AOR_MOVE/APG, too, now has an observed status value that indicates availability, but is different from STARTING.
16. This supplies the second startup prerequisite for CICSTOR/APL/SYSTEM3 (the one that is implicit in the HASPARENT relationship), so

CICSTOR/APL/SYSTEM3 is now started, too. It is of type APL rather than APG, so there is no further request propagation. To start CICSTOR/APL/SYSTEM3 means that the automation manager will send a start order for CICSTOR/APL/SYSTEM3 to the automation agent on SYSTEM3.

17. When CICSTOR/APL/SYSTEM3 is fully up and running, the automation agent on SYSTEM3 will report the fact back to the automation manager, which will set the *observed* status of CICSTOR/APL/SYSTEM3 to an observed status value that indicates availability, but is different from STARTING.
18. For the reasons given in 14 on page 78 and 16 on page 79, the observed status of CICSTOR/APL/SYSTEM3 is checked to determine when the BCKUPFIN event for CICSTOR/APL/SYSTEM3 will be unset.

How to Use User E-T Pairs

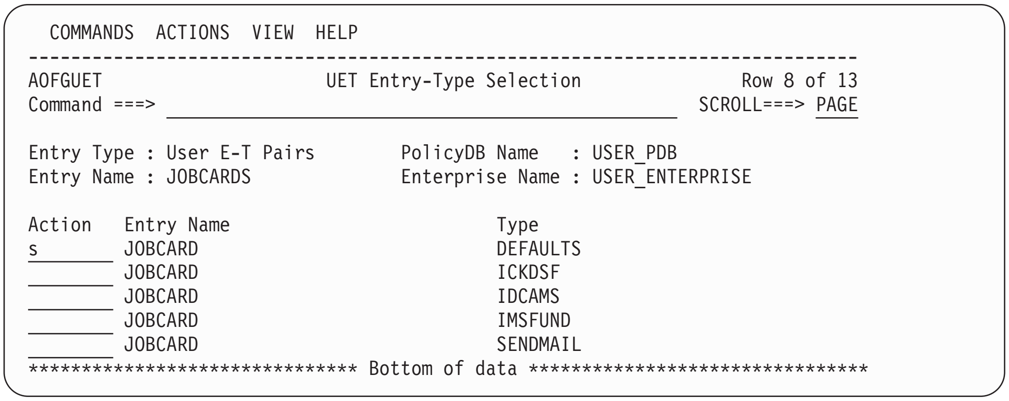
This section describes how user-specified data that does not fit into one of the standard entry types (for example, APL, APG, or SYS) can be entered into a policy database and retrieved later on by the automation.

In this example, the data for a job card is modified.

Step 1: Specifying User Data in the Customization Dialog

Select entry type User E-T Pairs (99) from the Entry Type Selection panel. In this example, assume that an entry JOBCARDS for a set of jobcards has already been created. Select this entry.

If you select the E-T DATA policy item from the Policy Selection panel for the JOBCARDS User E-T Pairs, the UET Entry-Type Selection panel similar to Figure 52 is displayed.

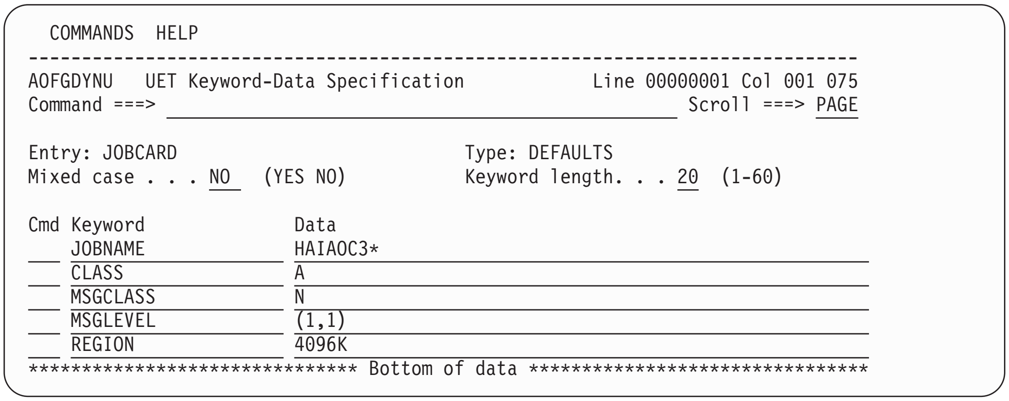


```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGUET                    UET Entry-Type Selection              Row 8 of 13
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : User E-T Pairs        PolicyDB Name   : USER_PDB
Entry Name : JOBCARDS              Enterprise Name : USER_ENTERPRISE

Action   Entry Name                         Type
s_______ JOBCARD                            DEFAULTS
________ JOBCARD                            ICKDSF
________ JOBCARD                            IDCAMS
________ JOBCARD                            IMSFUND
________ JOBCARD                            SENDMAIL
******************************* Bottom of data ********************************
```

Figure 52. UET Entry-Type Selection Panel

Enter S for Select in the **Action** column for the User E-T Pairs pair that you want to process. In this example, you want to process the JOBCARD of type DEFAULTS. SA z/OS displays the UET Keyword-Data Specification panel, where you can enter the keywords and user data for each Entry-Type pair data, as shown in Figure 53 on page 84. For more details about this panel, see “E-T DATA Policy Item” on page 313.

A keyword is an identifier for the data that follows it. The set of keyword/data pairs may be null, and each keyword does not have to be unique; that is, each keyword may have multiple sets of data assigned to it.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNU   UET Keyword-Data Specification           Line 00000001 Col 001 075
Command ===> ________________________________________________ Scroll ===> PAGE

Entry: JOBCARD                           Type: DEFAULTS
Mixed case . . . NO   (YES NO)           Keyword length. . . 20  (1-60)

Cmd Keyword              Data
___ JOBNAME              HAIAOC3*
___ CLASS                A
___ MSGCLASS             N
___ MSGLEVEL             (1,1)
___ REGION               4096K
******************************* Bottom of data ********************************
```

Figure 53. UET Keyword-Data Specification Panel

Step 2: Access the User Data

In a NetView session, you can use the command DISPACF to retrieve the data from the automation control file.

You can also retrieve the data using ACFFQRY from within a REXX exec as shown in the following example:

```
Get_keyword:   /* name of a subroutine in the REXX ecec */
   Arg keyword .
   len = Length(keyword)
   /* get the data within a pipe */
  'PIPE NetView ACFFQRY JOBCARD DEFAULTS | STEM ALL_DATA.',
     '| SEPARATE | LOCATE 1.'||len' /'||keyword||'/',
     '| STEM KEY.'
   If all_data.0 < 1 Then
     Call Terminal_Error 'PIPE 1 Failed'
   If all_data.1 <> 'ACFFQRY:0' Then
     Return
   If key.0 = 0 Then
     Say 'No Match for 'keyword
   Else Do i = 1 to key.0
     Say 'Match! 'key.i
   End

 Return
```

This method returns the following data in the pipe:

```
| IPSNO
ACFFQRY:0
JOBCARD DEFAULTS
JOBNAME=HAIAOC3*
CLASS=A
MSGCLASS=N
MSGLEVEL=(1,1)
REGION=4096K
```

After this processing, the data is available in the `key.` stem variable and you can use it for further processing.

How to Implement Message Processing with User-Defined Data

This section describes how user-defined data is used to automatically execute commands if an application issues a certain message.

Consider the following scenario:

- The HSM application issues the ARC0909E message that indicates the utilization percentage of the journal data set. You want to automate the recovery of the HSM journal data set if it is more than 97% full. Thus, you want SA z/OS to automatically issue two commands to HSM that will perform the recovery:

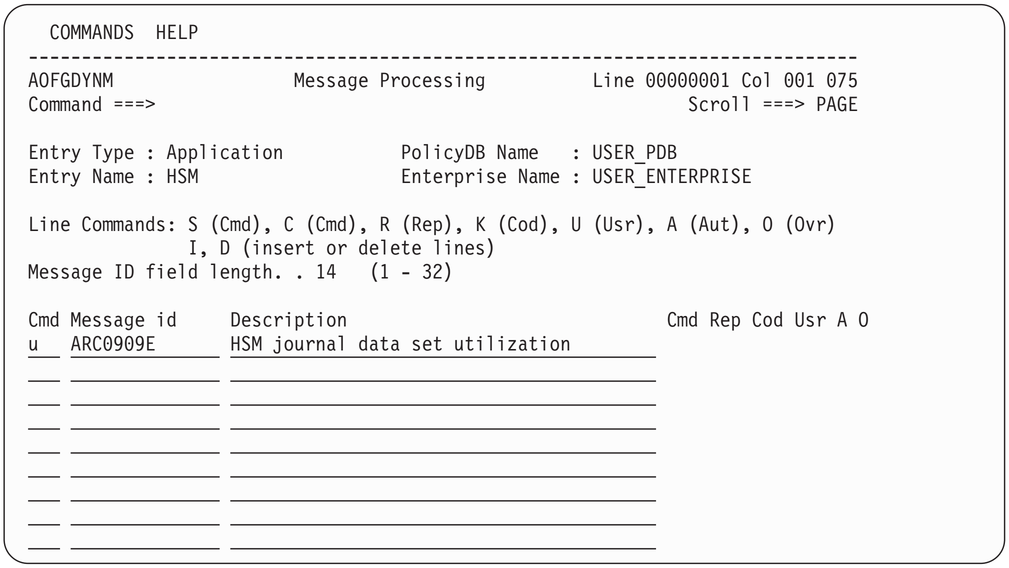
  HOLD ALL
  To stop all further activities for HSM.

  BACKVOL CDS
  To create a backup of the full journal data set and allocate a new one.

To implement this scenario, perform the steps described in the following sections.

Step 1: Specify the User-Defined Data

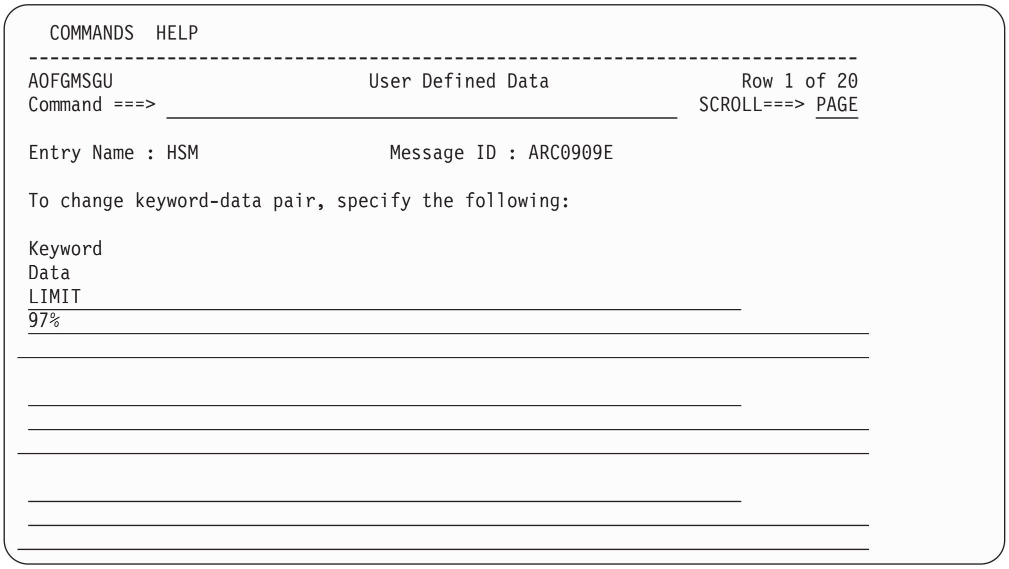
Select the application that you want to automate the recovery for and then select its policy item MESSAGES/USER DATA. This will lead you to panel Message Processing where you enter the message ID `ARC0909E` as shown in Figure 54.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNM                Message Processing         Line 00000001 Col 001 075
Command ===>                                                Scroll ===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : HSM                   Enterprise Name : USER_ENTERPRISE

Line Commands: S (Cmd), C (Cmd), R (Rep), K (Cod), U (Usr), A (Aut), O (Ovr)
               I, D (insert or delete lines)
Message ID field length. . 14   (1 - 32)

Cmd Message id     Description                                Cmd Rep Cod Usr A O
u   ARC0909E       HSM journal data set utilization
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
```

Figure 54. Message Processing Panel

To specify your data, type USR in the **Action** column of the Message Processing panel shown in Figure 54. Pressing Enter leads to the User Defined Data panel where you specify your input as shown in Figure 55 on page 86.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGU                       User Defined Data                  Row 1 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Name : HSM                 Message ID : ARC0909E

To change keyword-data pair, specify the following:

Keyword
Data
LIMIT
97%
```

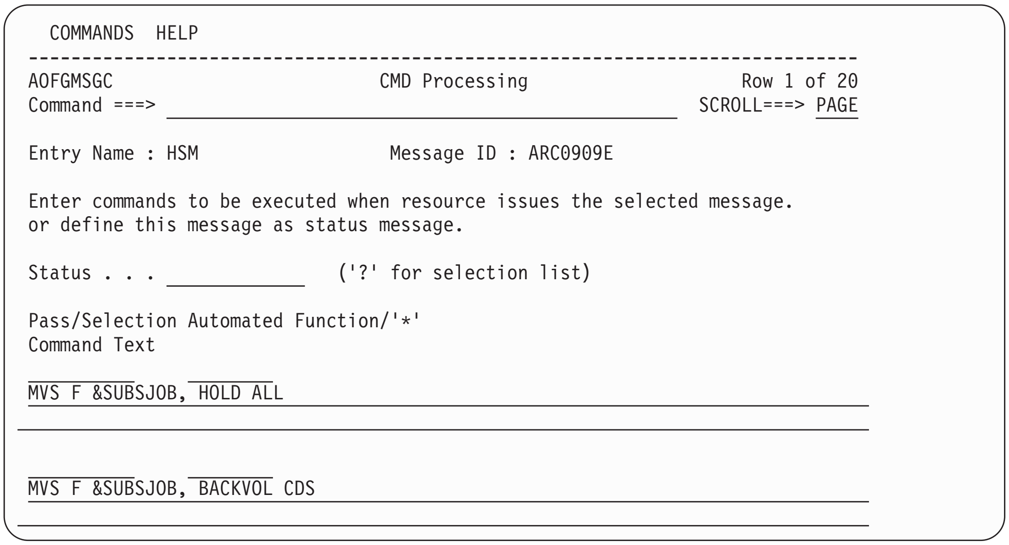
Figure 55. User Defined Data Panel

In the REXX exec that you will write in “Step 4: Write the REXX Exec” on page 88, you will compare the percentage value returned from the ARC0909E message with the value specified here, and, if it exceeds your specified value, you will cause SA z/OS to issue the commands specified in “Step 3: Update the NetView Automation Table” on page 87. If you later decide to start recovery if 95% usage is exceeded, you need not edit your REXX exec, but only your customization dialog input and then rebuild your system operations configuration files.

Pressing PF3 will store your input and bring you back to the Message Processing panel.

Step 2: Specify the Commands

In the Message Processing panel (Figure 54 on page 85) enter `CMD` in the **Cmd** column for message ARC0909E. This leads to the CMD Processing panel where you specify your input as shown in Figure 56.



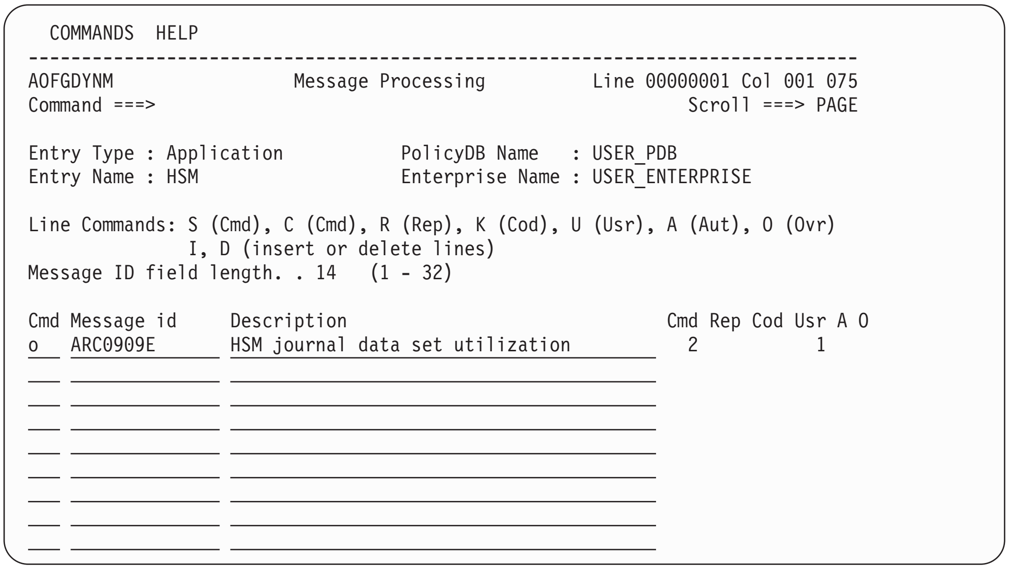
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGC                        CMD Processing                    Row 1 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Name : HSM                 Message ID : ARC0909E

Enter commands to be executed when resource issues the selected message.
or define this message as status message.

Status . . .              ('?' for selection list)

Pass/Selection Automated Function/'*'
Command Text

MVS F &SUBSJOB, HOLD ALL


MVS F &SUBSJOB, BACKVOL CDS
```

Figure 56. CMD Processing Panel

Step 3: Update the NetView Automation Table

You need to specify a NetView automation table entry to trigger a REXX exec called YOUR_CLIST, which you will create in “Step 4: Write the REXX Exec” on page 88.

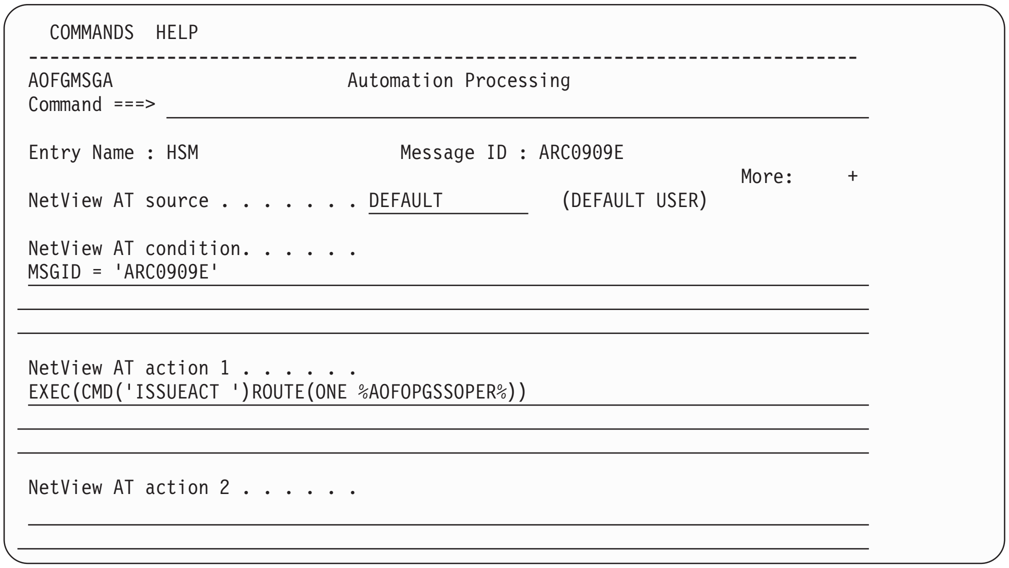
The AT entry can be specified by using the *Automation Processing* panel to override the predefined AT entry. To do this, enter OVR in the **Action** column for message ARC0909E in the *Message Processing* panel, as shown in Figure 57.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNM                 Message Processing          Line 00000001 Col 001 075
Command ===>                                                  Scroll ===> PAGE

Entry Type : Application            PolicyDB Name   : USER_PDB
Entry Name : HSM                    Enterprise Name : USER_ENTERPRISE

Line Commands: S (Cmd), C (Cmd), R (Rep), K (Cod), U (Usr), A (Aut), O (Ovr)
              I, D (insert or delete lines)
Message ID field length. . 14   (1 - 32)

Cmd Message id     Description                              Cmd Rep Cod Usr A O
o   ARC0909E       HSM journal data set utilization           2           1
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
___ ______________ ________________________________________
```

Figure 57. Message Processing Panel

This leads to the Automation Processing panel, as shown in Figure 58. This shows the default values for the message.

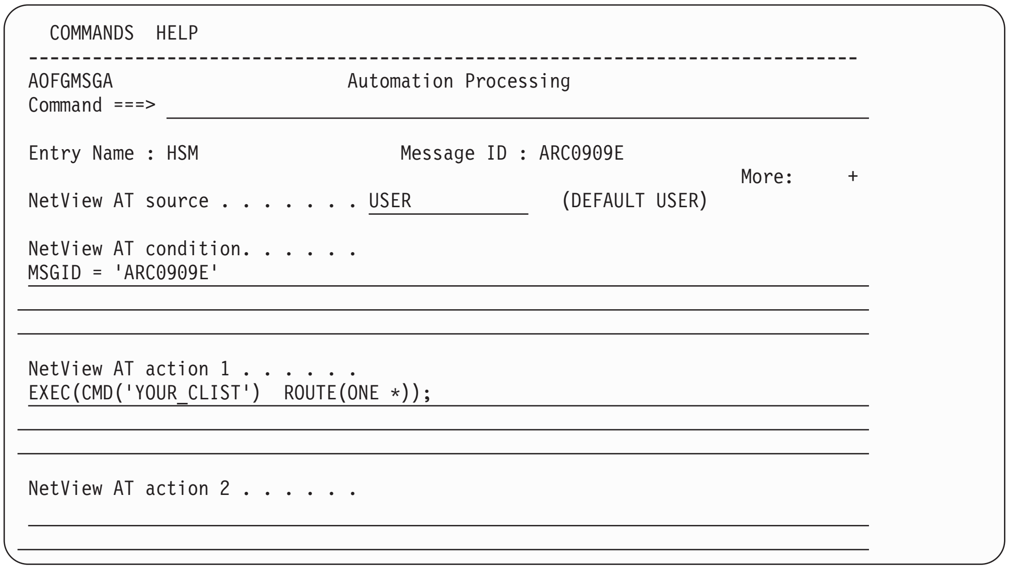


```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGA                      Automation Processing
Command ===> __________________________________________________________________

Entry Name : HSM                    Message ID : ARC0909E
                                                                More:     +
NetView AT source . . . . . . . DEFAULT           (DEFAULT USER)

NetView AT condition. . . . . .
MSGID = 'ARC0909E'
________________________________________________________________________________
________________________________________________________________________________

NetView AT action 1 . . . . . .
EXEC(CMD('ISSUEACT ')ROUTE(ONE %AOFOPGSSOPER%))
________________________________________________________________________________
________________________________________________________________________________

NetView AT action 2 . . . . . .
________________________________________________________________________________
________________________________________________________________________________
```

Figure 58. Automation Processing Panel

The definitions that are needed to create an AT entry are as shown in the action fields in Figure 59 on page 88. Pressing Enter saves your override and the fields are highlighted in a different color, and the **NetView AT source** field is changed to

USER to indicate the override.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGA                     Automation Processing
Command ===> ____________________________________________________________________

Entry Name : HSM                  Message ID : ARC0909E
                                                                  More:     +
NetView AT source . . . . . . . USER               (DEFAULT USER)

NetView AT condition. . . . . .
MSGID = 'ARC0909E'



NetView AT action 1 . . . . . .
EXEC(CMD('YOUR_CLIST')  ROUTE(ONE *));



NetView AT action 2 . . . . . .


```

Figure 59. Automation Processing Panel

Alternatively, you can manually specify this automation table entry either in INGMSGU1 (which is empty, but already included in INGMSG01), or in your own automation table, which then needs to be specified in the AUTOMATION SETUP policy for entry type SYS.

This entry should look like the following:

```
IF MSGID = 'ARC0909E'
THEN EXEC(CMD('YOUR_CLIST')  ROUTE(ONE *));
```

Step 4: Write the REXX Exec

The value specified on the User Defined Data panel (Figure 55 on page 86) is accessed in the PIPE statement using the ACFFQRY command and stored in the variable limit (1).

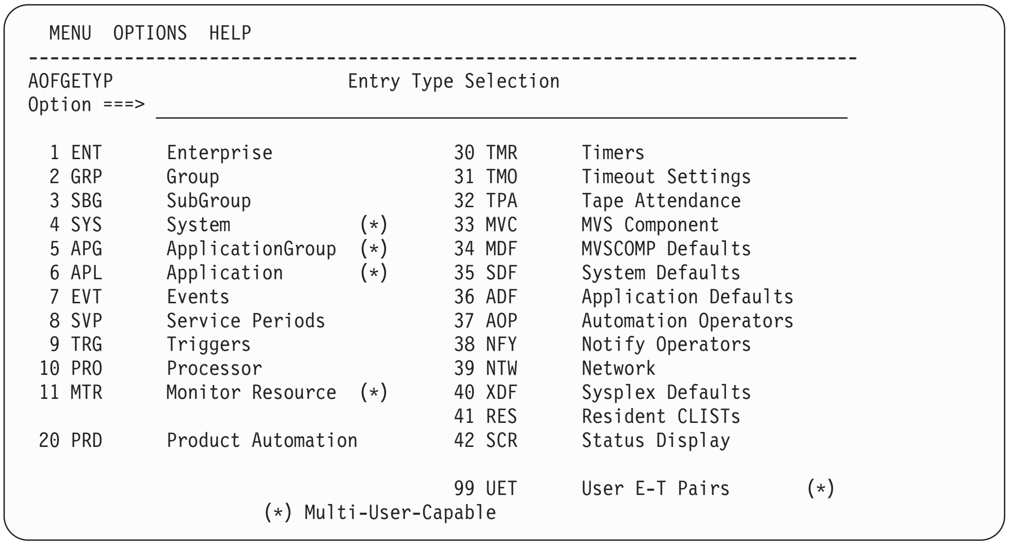
To issue the commands specified on the CMD Processing panel (Figure 56 on page 86), the ACFCMD command is used (2).

```
/* REXX */

/* debugging ... */
.....

lrc = 0

/* Get the message from the safe */
"GETMLINE txt 1"

svmsgid = msgid()
svjobname = jobname()
Parse Var txt . type . ' IS ABOUT ' usage '% ' .

/* This sample recovers just type JRNL */
If type \= 'JRNL' Then Exit 0

/* Check the RECOVERY automation flag */
"AOCQRY "svjobname"."svmsgid" RECOVERY"
lrc = rc
```

```
Select
  /* Automation allowed */
  When lrc = 0 Then Do
    /* Get TGlobal variable from AOCQRY */
    "GLOBALV GETT SUBSAPPL"

    /* Get the user-data from the ACF */
    "PIPE (STAGESEP | NAME GETUDATA)" ,
      "NETV ACFFQRY "subsappl","svmsgid ,  1
      "| SEPARATE" ,
      "| LOCATE 1.6 /LIMIT=/" ,
      "| EDIT 7.2 1" ,
      "| VAR limit"

    /* If no limit was defined use default */
    If symbol('limit') = 'LIT' Then limit = '95'

    /* Check if limit is exeeded. If so, issue commands */
    If usage >= limit Then Do
      "ACFCMD MSGTYP="svmsgid              2
      lrc = rc

      /* Error occured ? */
      If lrc \= 0 Then Do
        ...
      End
  End
  /* Automation not allowed */
  When lrc < 3 Then Do
    ...
  End
  /* Resource not known to SA */
  When lrc = 3 Then Nop
  /* Error occured */
  Otherwise Do
    ...
  End
End

Exit lrc
```

Chapter 5. Policy Object Reference

This chapter describes how to define SA z/OS policy objects. They are described in the following sections in the order they appear on the Entry Type Selection panel shown in Figure 60.



```
  MENU  OPTIONS  HELP
------------------------------------------------------------------------------
AOFGETYP                      Entry Type Selection
Option ===> _________________________________________________________________

  1 ENT      Enterprise                30 TMR      Timers
  2 GRP      Group                     31 TMO      Timeout Settings
  3 SBG      SubGroup                  32 TPA      Tape Attendance
  4 SYS      System            (*)     33 MVC      MVS Component
  5 APG      ApplicationGroup  (*)     34 MDF      MVSCOMP Defaults
  6 APL      Application       (*)     35 SDF      System Defaults
  7 EVT      Events                    36 ADF      Application Defaults
  8 SVP      Service Periods           37 AOP      Automation Operators
  9 TRG      Triggers                  38 NFY      Notify Operators
 10 PRO      Processor                 39 NTW      Network
 11 MTR      Monitor Resource  (*)     40 XDF      Sysplex Defaults
                                       41 RES      Resident CLISTs
 20 PRD      Product Automation        42 SCR      Status Display

                                       99 UET      User E-T Pairs       (*)
                    (*) Multi-User-Capable
```

Figure 60. Entry Type Selection Panel

This chapter discusses the following policy objects:

- “Enterprise Policy Object” on page 92
- “Group Policy Object” on page 99
- “SubGroup Policy Object” on page 116
- “System Policy Object” on page 118
- “ApplicationGroup Policy Object” on page 134
- “Application Policy Object” on page 156
- “Events Policy Object” on page 233
- “Service Periods Policy Object” on page 235
- “Triggers Policy Object” on page 238
- “Processor Policy Object” on page 241
- “Monitor Resource Policy Object” on page 249
- “Product Automation Policy Object” on page 259
- “Timers Policy Object” on page 274
- “Timeout Settings Policy Object” on page 277
- “Tape Attendance Policy Object” on page 279
- “MVS Component Policy Object” on page 282
- “MVSCOMP Defaults Policy Object” on page 287
- “System Defaults Policy Object” on page 288
- “Application Defaults Policy Object” on page 290
- “Automation Operators Policy Object” on page 291
- “Notify Operators Policy Object” on page 296

- “Network Policy Object” on page 299
- “Sysplex Defaults Policy Object” on page 307
- “Resident CLISTs Policy Object” on page 309
- “Status Display Policy Object” on page 310
- “User E-T Pairs Policy Object” on page 313

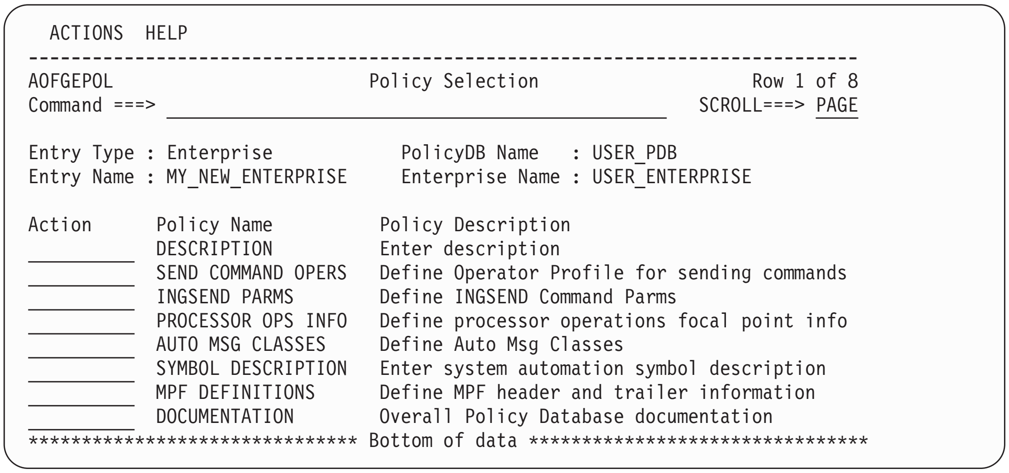
Enterprise Policy Object

You can define one enterprise for each policy database. An enterprise definition includes:

- Enterprise description
- operator profile and command parameters for INGSEND
- Processor operations focal point information

Policy Items for an Enterprise

Figure 61 shows the policy items that are displayed when *Enterprise* is selected from the Entry Type Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                    Row 1 of 8
Command ===> ______________________________________________   SCROLL===> PAGE

Entry Type : Enterprise            PolicyDB Name   : USER_PDB
Entry Name : MY_NEW_ENTERPRISE     Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  SEND COMMAND OPERS   Define Operator Profile for sending commands
__________  INGSEND PARMS        Define INGSEND Command Parms
__________  PROCESSOR OPS INFO   Define processor operations focal point info
__________  AUTO MSG CLASSES     Define Auto Msg Classes
__________  SYMBOL DESCRIPTION   Enter system automation symbol description
__________  MPF DEFINITIONS      Define MPF header and trailer information
__________  DOCUMENTATION        Overall Policy Database documentation
******************************* Bottom of data ********************************
```

Figure 61. Policy Selection Panel for Enterprises

The policy items for Enterprises are:

SEND COMMAND OPERS
This policy item allows you to define default RMTCMD autotasks for operators sending commands to target domains, so that they do not have to remember which target operator ID to use for a particular domain. This gives you more flexibility than the RMTCMD default of taking the task ID of the issuing operator.

These definitions are also used for cross domain command routing to systems that are outside the local sysplex. This is valid for all operator commands supporting the TARGET parameter.

For more information, see “SEND COMMAND OPERS Policy Item” on page 93.

INGSEND PARMS
This policy item allows you to define default parameters for the INGSEND command.

For more information refer to “INGSEND PARMS Policy Item” on page 95.

PROCESSOR OPS INFO
This policy item allows you to certain enterprise-wide processor operations data. For more information, refer to “PROCESSOR OPS INFO Policy Item” on page 96.

AUTO MSG CLASSES
This policy item allows you to define your own message classes. Message classes that are defined here can be assigned to operators by using policy NOTIFY OPERATORS of the entry type NFY. For more information, see “AUTO MSG CLASSES Policy Item” on page 96

SYMBOL DESCRIPTION
This policy item allows you to enter system automation symbol descriptions. For more information, see “SYMBOL DESCRIPTION Policy Item” on page 97

MPF DEFINITIONS
This policy item allows you to define MPF header and trailer information. For more information, see “MPF DEFINITIONS Policy Item” on page 98

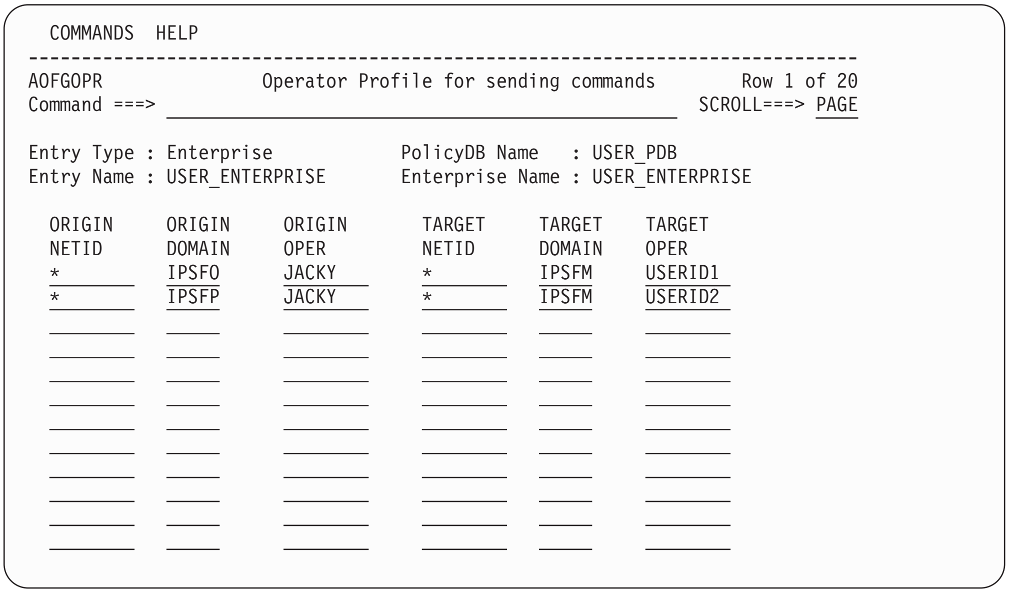
DOCUMENTATION
This policy item allows you to enter or update any text with the ISPF editor. You can use it to:

- Enter overall policy database descriptions
- Look at documentation that is shipped together with predefined SA z/OS sample add-on policies

Note: The policy database is locked for updates by other users while you are editing data in this policy item.

SEND COMMAND OPERS Policy Item

Figure 62 shows the panel that is displayed when the SEND COMMAND OPERS policy item is selected.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGOPR              Operator Profile for sending commands        Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Enterprise             PolicyDB Name   : USER_PDB
Entry Name : USER_ENTERPRISE        Enterprise Name : USER_ENTERPRISE

  ORIGIN     ORIGIN     ORIGIN      TARGET     TARGET     TARGET
  NETID      DOMAIN     OPER        NETID      DOMAIN     OPER
  *_______   IPSFO      JACKY___    *_______   IPSFM      USERID1_
  *_______   IPSFP      JACKY___    *_______   IPSFM      USERID2_
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
  ________   _____      ________    ________   _____      ________
```

Figure 62. Operator Profile for Sending Commands Panel

This policy item allows you to define default RMTCMD autotasks for operators sending commands to target domains, so that they do not have to remember which target operator ID to use for a particular domain. This gives you more flexibility than the RMTCMD default of taking the task ID of the issuing operator.

These definitions are also used for cross domain command routing to systems that are outside the local sysplex. This is valid for all operator commands supporting the TARGET parameter.

This policy item is intended to help you migrate your cross domain command routing to RMTCMD, by allowing you to use any naming scheme for RMTCMD autotasks. By default, RMTCMD uses the operator ID of the invoking task to determine the RMTCMD autotask to issue a command to. Use the Operator Profile for Sending Commands panel to define an association between an origin operator and a target RMTCMD autotask. The INGSEND command uses predefined RMTCMD associations to resolve the OPER keyword (target operator ID to execute the routed command), if it has not been specified on an invocation. If INGSEND does not find an RMTCMD association for the invoking operator to the target domain in the automation control file, the RMTCMD default will be used to resolve the target operator ID. For more information on INGSEND, refer to *IBM Tivoli System Automation for z/OS User's Guide* and *IBM Tivoli System Automation for z/OS Operator's Commands*. For more information on RMTCMD, refer to *Tivoli NetView for z/OSCommand Reference Volume 1*.

The definitions you enter here apply to all systems in the enterprise. They are included in the automation control file built for each system. At run time, INGSEND on each origin domain will use only the information intended for that particular origin Netid.Domain.

On the Operator Profile for Sending Commands panel (Figure 62 on page 93), the following fields are defined for both origin and target:

NETID
The Network ID is optional. The maximum length is eight characters. The first character must be alphanumeric. The rest must be either alphanumeric or national characters.

DOMAIN
The NetView Domain ID is optional. The maximum length is five characters. The first character must be alphanumeric. The rest must be either alphanumeric or national characters.

OPER The Origin Operator ID is required. The maximum length is eight characters.

The Target Operator ID is optional. If entered, it must be eight characters or less. If the field is left blank, it will resolve to the ORIGIN OPER value.

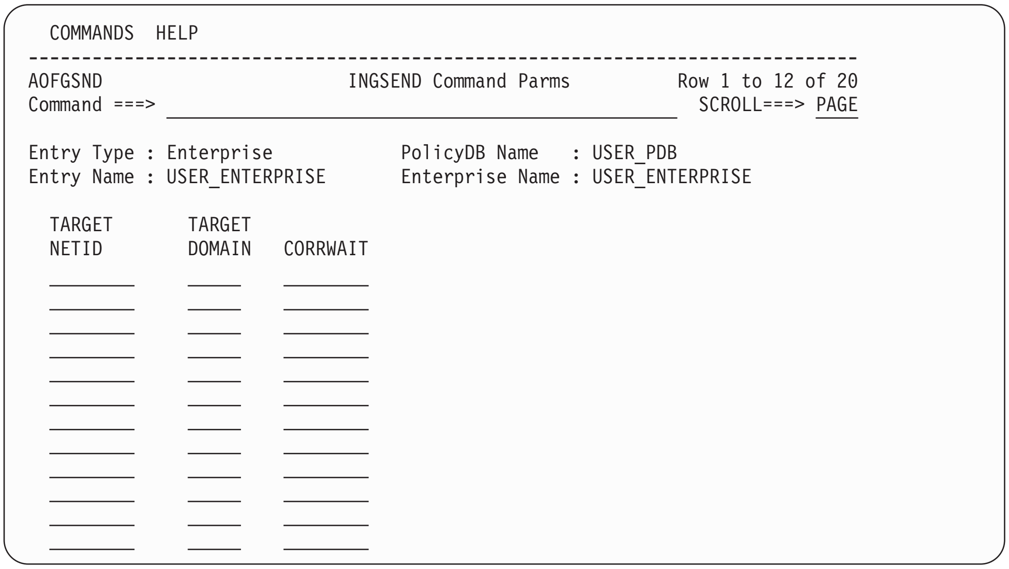
The operator ID you define here should have matching OPERATOR statements in the relevant DSIOPF for each origin and target domain.

> **Note:**
> For TARGET OPER, PPT or target domain concatenated with PPT is not allowed.

A wildcard character, *, can be used for NETID and DOMAIN. All fields are optional except for ORIGIN OPER. TARGET OPER will resolve to ORIGIN OPER if it is not entered. This is the NetView default.

INGSEND PARMS Policy Item

If you select policy item INGSEND PARMS from the Policy Selection panel for enterprises, a panel similar to Figure 63 is shown.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGSND                       INGSEND Command Parms         Row 1 to 12 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Enterprise            PolicyDB Name   : USER_PDB
Entry Name : USER_ENTERPRISE       Enterprise Name : USER_ENTERPRISE

  TARGET       TARGET
  NETID        DOMAIN   CORRWAIT
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
  ________     _____    ________
```

Figure 63. INGSEND Command Parms Panel

The System Operations INGSEND command is used to route commands across domains. Only one command routing mechanism is supported: the NetView® RMTCMD command.

RMTCMD-based INGSEND invocations can involve usage of the NetView PIPE command. Because RMTCMD produces asynchronous command output, it is necessary to use the CORRWAIT Pipe stage to trap the asynchronous messages. You can customize the CORRWAIT time that INGSEND uses per target domain.

Use wildcards for target netid and domain to specify the predominant CRM in your enterprise.

TARGET NETID
: The Target Network ID is optional and, if entered, must be eight characters or less. The first character must be alphanumeric. The rest must be either alphanumeric or national characters. If left blank, it defaults to the wildcard value of ’*’.

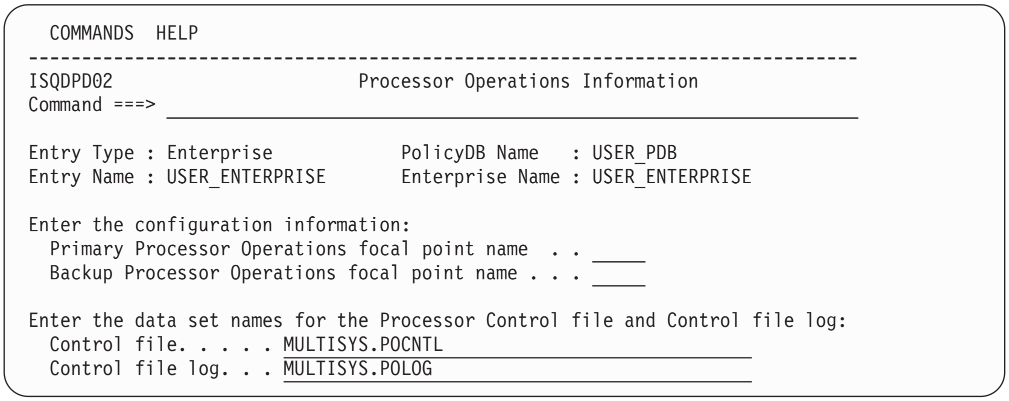
TARGET DOMAIN
: The Target Domain is optional and, if entered, must be five characters or less. The first character must be alphanumeric. The rest must be either alphanumeric or national characters. If left blank, it defaults to the wildcard value of ’*’.

Wildcard value of ’*’ is allowed.

CORRWAIT
: The corrwait value is optional. This value is used by INGSEND for the NetView PIPE CORRWAIT stage. If left blank, it defaults to 10 seconds.

PROCESSOR OPS INFO Policy Item

A panel as shown in Figure 64 is displayed if you select policy item PROCESSOR OPS INFO from the Policy Selection panel for enterprises.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
ISQDPD02                       Processor Operations Information
Command ===> __________________________________________________________________

Entry Type : Enterprise            PolicyDB Name   : USER_PDB
Entry Name : USER_ENTERPRISE       Enterprise Name : USER_ENTERPRISE

Enter the configuration information:
  Primary Processor Operations focal point name  . . _____
  Backup Processor Operations focal point name . . . _____

Enter the data set names for the Processor Control file and Control file log:
  Control file. . . . . MULTISYS.POCNTL____________________________
  Control file log. . . MULTISYS.POLOG_____________________________
```

Figure 64. Processor Operations Information Panel

The panel contains the following fields:

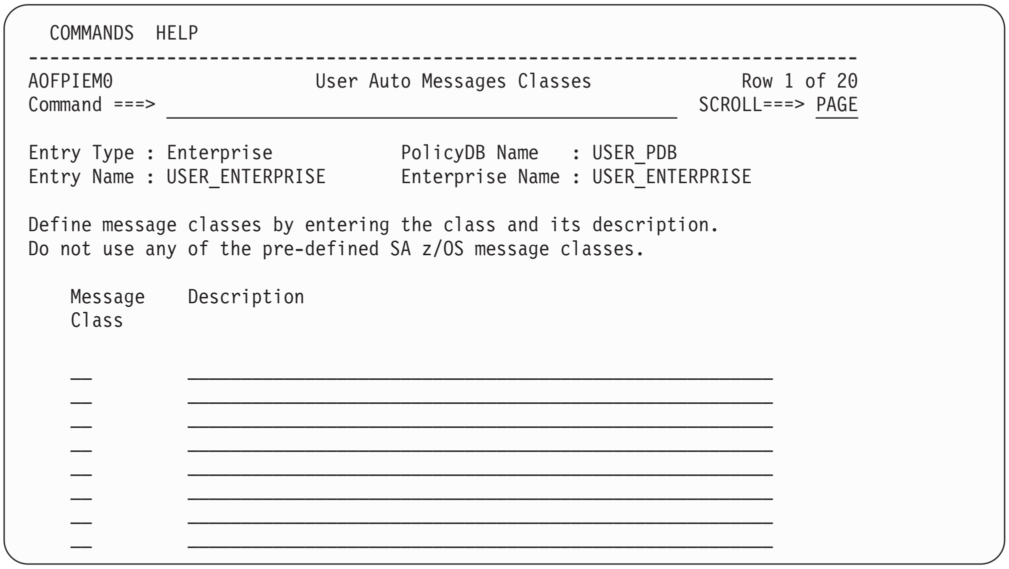
Focal point name (primary or backup)
: Enter the NetView domain name of the NetView that is the name of the focal point for your SA z/OS configuration. Names can contain 5 characters, starting with an alphabetic character, #, $, or @. This field is required if you are using SA z/OS processor operations functions. The customization dialog does not verify your entry.

Control file
: Identifies the data set containing configuration information by processor operations under NetView.

Control file log
: Identifies the data set containing information about errors in the configuration definition. When you generate a control file, this error file contains the errors or messages created during the generation procedure. Check this file after each control file generation.

AUTO MSG CLASSES Policy Item

A panel as shown in Figure 65 on page 97 is displayed if you select policy item AUTO MSG CLASSES on the Policy Selection panel for enterprises.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIEM0                 User Auto Messages Classes             Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Enterprise           PolicyDB Name   : USER_PDB
Entry Name : USER_ENTERPRISE      Enterprise Name : USER_ENTERPRISE

Define message classes by entering the class and its description.
Do not use any of the pre-defined SA z/OS message classes.

    Message    Description
    Class

    __         _______________________________________________________
    __         _______________________________________________________
    __         _______________________________________________________
    __         _______________________________________________________
    __         _______________________________________________________
    __         _______________________________________________________
    __         _______________________________________________________
    __         _______________________________________________________
```

Figure 65. AUTO MSG CLASSES Panel

This panel defines the user message classes that are available to be assigned to notify operators by using policy NOTIFY OPERATORS of entry type NFY. The Auto Msg Classes policy object allows you to define your own message classes.

You can also enter a short description of each message class. Message classes must consist of an alpha and a numeric character. All double numeric message classes are reserved for use by IBM. See “Notify Operators Policy Object” on page 296 for more information on the message classes that SA z/OS defines.

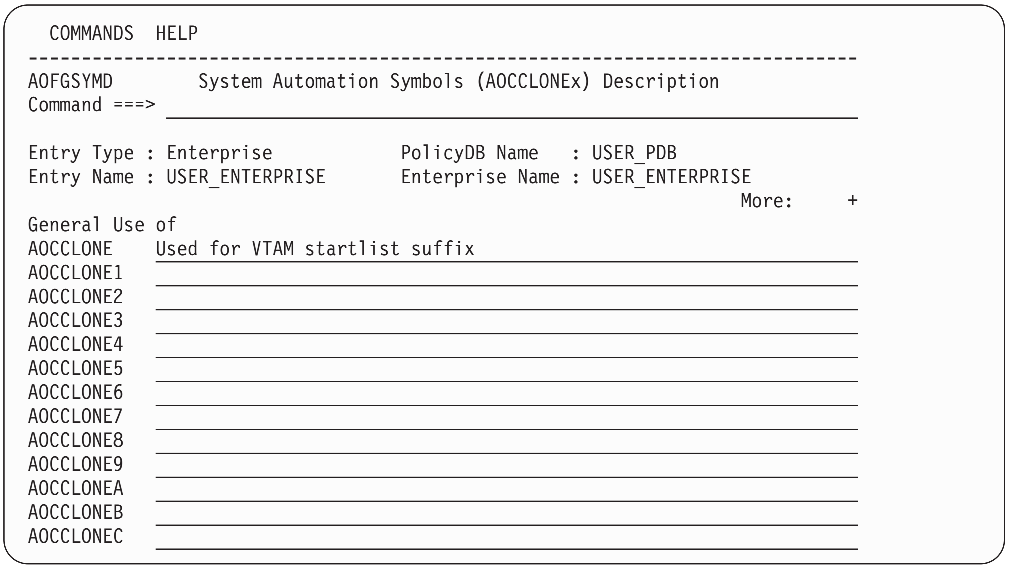
Notes:

1. No duplicate message classes are allowed.
2. Message classes used within any Notify Operators cannot be deleted.

For details about the definition restrictions of message classes, refer to the online help of SA z/OS.

SYMBOL DESCRIPTION Policy Item

If you select policy item SYMBOL DESCRIPTION from the Policy Selection panel for enterprises, a panel similar to Figure 66 on page 98 is shown.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGSYMD        System Automation Symbols (AOCCLONEx) Description
Command ===>

Entry Type : Enterprise           PolicyDB Name   : USER_PDB
Entry Name : USER_ENTERPRISE      Enterprise Name : USER_ENTERPRISE
                                                              More:     +
General Use of
AOCCLONE    Used for VTAM startlist suffix
AOCCLONE1
AOCCLONE2
AOCCLONE3
AOCCLONE4
AOCCLONE5
AOCCLONE6
AOCCLONE7
AOCCLONE8
AOCCLONE9
AOCCLONEA
AOCCLONEB
AOCCLONEC
```

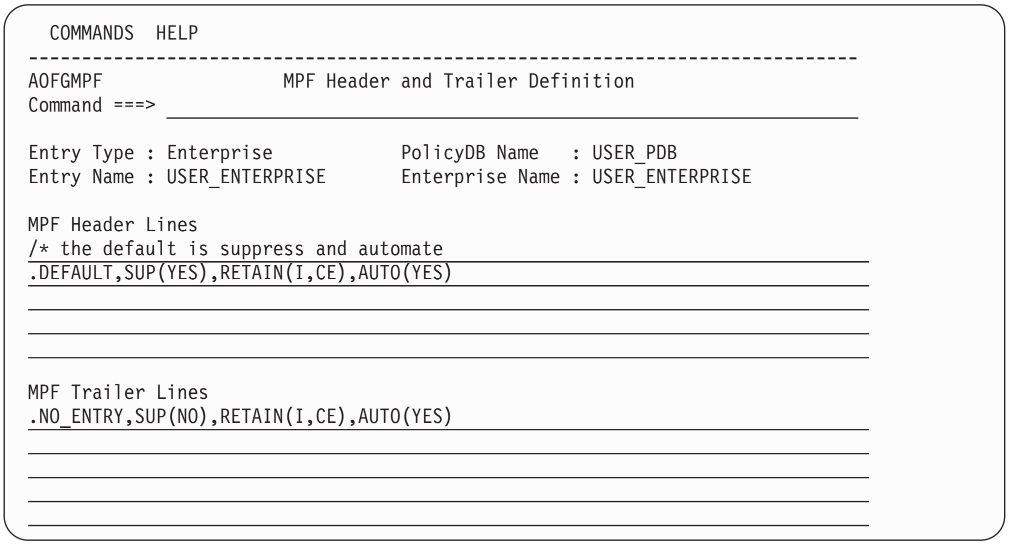
Figure 66. System Automation Symbols (AOCCLONEx) Description Panel

This panel allows you to enter the description of the general use of each system automation symbol.

AOCCLONE*x*
: This allows you to describe the general use of each system automation symbol. This description is shown in the AUTOMATION SYMBOLS policy of all systems (entry type SYS) in the policy database.

MPF DEFINITIONS Policy Item

If you select policy item MPF DEFINITIONS from the Policy Selection panel for enterprises, a panel similar to Figure 67 is shown.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMPF                MPF Header and Trailer Definition
Command ===>

Entry Type : Enterprise           PolicyDB Name   : USER_PDB
Entry Name : USER_ENTERPRISE      Enterprise Name : USER_ENTERPRISE

MPF Header Lines
/* the default is suppress and automate
.DEFAULT,SUP(YES),RETAIN(I,CE),AUTO(YES)




MPF Trailer Lines
.NO_ENTRY,SUP(NO),RETAIN(I,CE),AUTO(YES)
```

Figure 67. MPF Header and Trailer Definition Panel

This panel allows you to enter or update definitions which will be included in the MPFLSTSA member during MPF build.

MPF Header Lines

Allows you to specify up to 5 MPF statements that will be included at the top of the MPFLSTSA member, for example:

```
.NO_ENTRY,SUP(NO),RETAIN(I,CE),AUTO(YES)
```

or

```
.DEFAULT,SUP(YES),RETAIN(I,CE),AUTO(YES)
```

Comment lines may be specified by including a / followed by an * as the first 2 characters.

MPF Trailer Lines

Allows you to specify up to 3 MPF statements that will be included at the bottom of the MPFLSTSA member.

Note:

Changes in MPF header and trailer lines are taken into account during the *next* build of NetView automation tables when the MPF member is also rebuilt.

Group Policy Object

A group is the largest subdivision of an SA z/OS enterprise. It is a set of systems or subgroups, or both, that you want to associate with one another for purposes of automation, monitoring on the NMC workstation or that you want to use with the SA z/OS processor operations functions. An SA z/OS enterprise may have many groups, each consisting of one or more systems and subgroups. You define a group by assigning a name to it and by selecting the member systems and subgroups.

The status of a group depends on the status of the systems or applications that it is comprised of. The status of a group is indicated by appropriate colors on the NMC workstation. You specify the number of systems or applications of a group that must be in an unsatisfactory state to cause SA z/OS to change the color of a group in the graphic display.

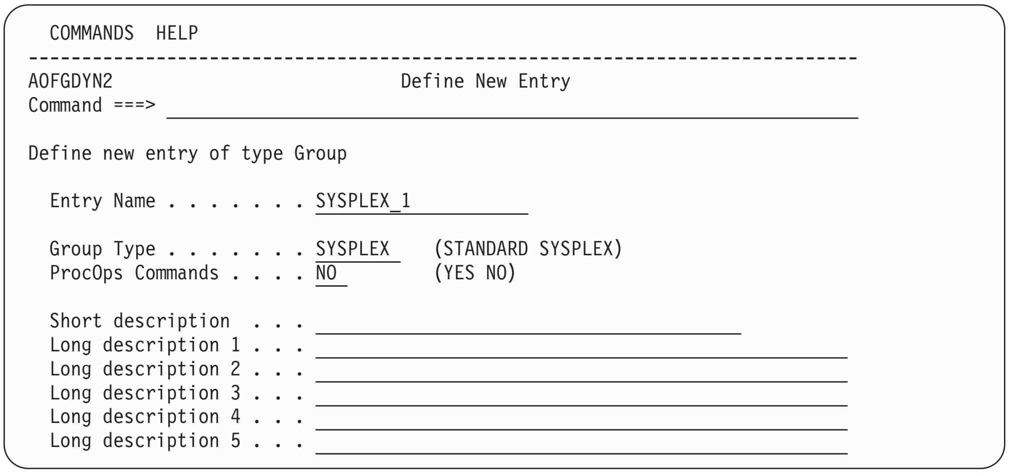
Creating a New Group

How to create new policy objects in general, is described in “How to Create New Policy Objects” on page 37.

Groups can be defined as either *standard groups* or *sysplex groups*. A standard group contains any combination of systems and subgroups. A sysplex group contains only systems or subgroups that belong to the same XCF group. For a sysplex group, there are additional policy items to describe certain sysplex-specific resources. A sysplex group may also have Sysplex Application Groups connected to it. Sysplex Application Groups are groups of applications that can be located on more than one system in the sysplex. A group must be defined as either a sysplex group or a standard group when it is created. Figure 68 on page 100 is an example of the Define New Entry panel for groups.

A system may belong to many standard groups, but can belong to only one sysplex group.

If you want to monitor your newly created sysplex group on the NMC workstation, you need to specify an entry for this sysplex in the INGTOPOF file (see *IBM Tivoli System Automation for z/OS Planning and Installation*).



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN2                           Define New Entry
Command ===> _________________________________________________________________

Define new entry of type Group

  Entry Name . . . . . . . SYSPLEX_1___________

  Group Type . . . . . . . SYSPLEX_   (STANDARD SYSPLEX)
  ProcOps Commands . . . . NO_        (YES NO)

  Short description  . . . ________________________________________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 68. Define New Entry Panel for a Sysplex Group

The unique fields are:

Name
This is the name of the entry as used by the SA z/OS dialogs. Each Group must be assigned a unique, non-blank name. The name must be unique among the names of all groups, subgroups, systems (including both the entry names and the Image/ProcOps names), and processors in the current policy database.

Note:
This name can be used in the TARGET parameter of SA z/OS commands. If the specified name is longer than 8 characters, truncation of the name may occur on certain SA z/OS fullscreen command panels.

When running in local mode (sysplex() function returns LOCAL) you must define a Group name if you want to use SA z/OS commands that target other systems via the TARGET parameter.

Group type
Use this field to specify the type of group you want to create. You can create either a Sysplex group if the group is to be associated with a sysplex, or a Standard group. Each type of group has specific policy items and restrictions associated with it. The value you enter here cannot be changed after the group is created.

The available values are:

SYSPLEX
If the Group is being used to represent a logical sysplex (a group of systems within the same XCF group ID).

STANDARD
For all other Groups, for example, to represent a physical sysplex (a group of systems within different XCF group IDs, such as a GDPS® site).

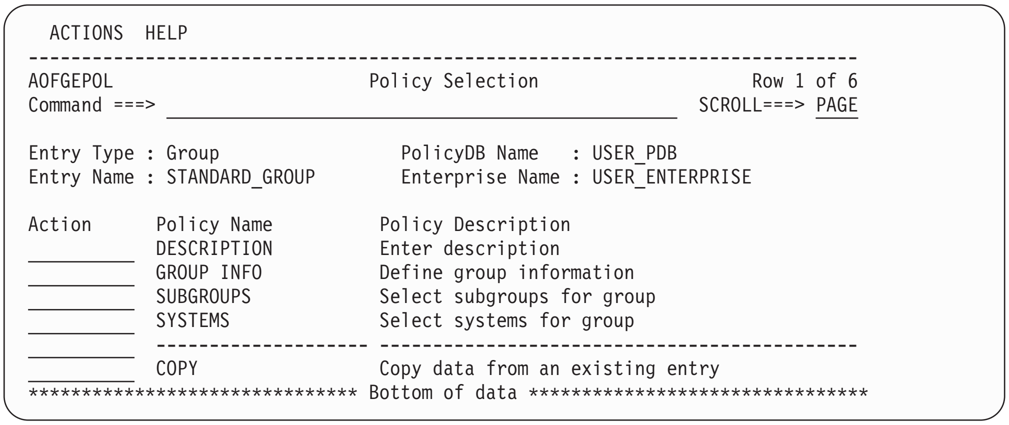
ProcOps Commands
Indicates whether processor operations commands issued against this

group are to be accepted for processing. The default is NO. See *IBM Tivoli System Automation for z/OS Operator's Commands* for information about the processor operations commands.

Note:
If you specify YES, a processor operations command directed to that group will affect all systems in that group, even if those systems are associated with a subgroup of the group and that subgroup is **not** enabled for processor operations commands.

Policy Items for Standard Groups

Figure 69 shows the policy items that are displayed for a standard group.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                    Row 1 of 6
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : STANDARD_GROUP        Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  GROUP INFO           Define group information
__________  SUBGROUPS            Select subgroups for group
__________  SYSTEMS              Select systems for group
__________  -------------------- --------------------------------------------
__________  COPY                 Copy data from an existing entry
****************************** Bottom of data ********************************
```

Figure 69. Policy Selection Panel for Standard Groups

The unique policy items for groups are:

GROUP INFO
This displays the type of group that was specified when the group was created. This can be either standard or sysplex. It also displays and allows you to update the current value of the processor operations commands indicator.

For more information, see "GROUP INFO Policy Item" on page 104.

SUBGROUPS
When you select this policy item, a list of all subgroups in the SA z/OS enterprise is displayed. Those subgroups that are related to the current group are marked with SELECT. You can add subgroups to the group, or remove subgroups from the group.

For more information, see "SUBGROUPS Policy Item" on page 104.

SYSTEMS
When you select this policy item, it displays a list of all systems in the SA z/OS enterprise. `SELECTED` in the **Status** column indicates the systems that have been linked to the group. You can add systems to or remove systems from the group.

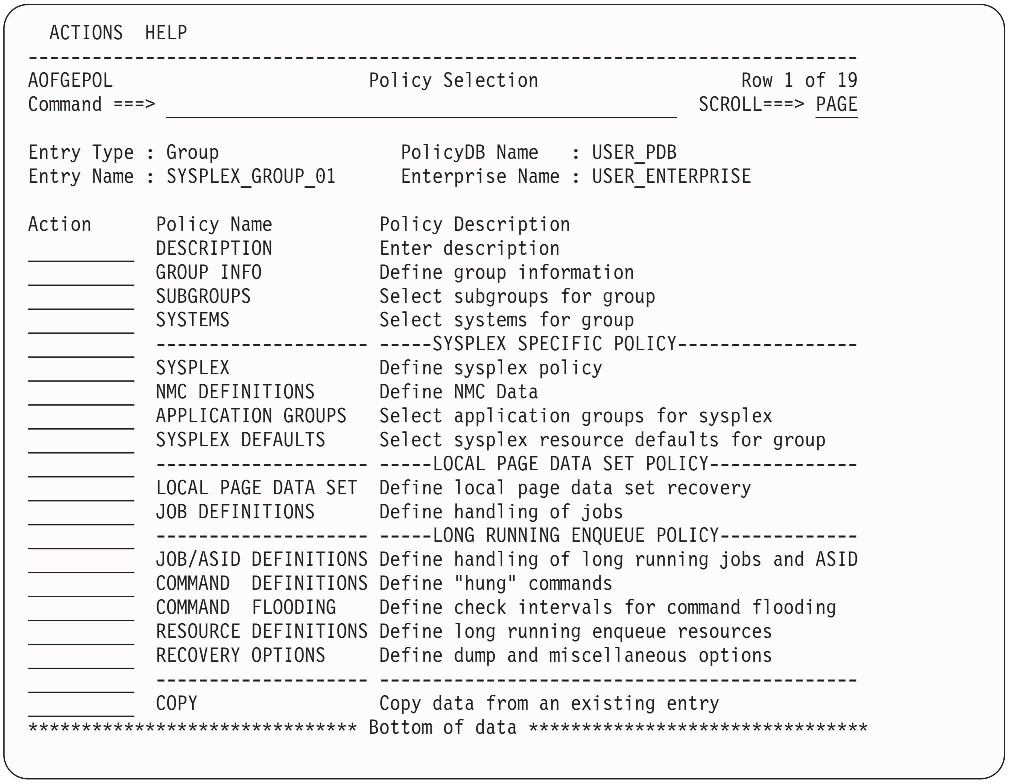
For more information, see "SYSTEMS Policy Item" on page 105.

Policy Items for Sysplex Groups

You define a sysplex group when creating a new group object by specifying `sysplex` in the **Group Type** field of the Define New Entry panel. There are certain restrictions for sysplex groups:

- A system can be linked to only one sysplex group. The same system can be linked to other standard groups. This applies if the system is linked directly to the system group, or indirectly through a subgroup.
- A subgroup may be linked to only one sysplex group.

Figure 70 is an example of the Policy Selection panel for Sysplex groups.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                 Row 1 of 19
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : SYSPLEX_GROUP_01      Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  GROUP INFO           Define group information
__________  SUBGROUPS            Select subgroups for group
__________  SYSTEMS              Select systems for group
__________  -------------------- -----SYSPLEX SPECIFIC POLICY-----------------
__________  SYSPLEX              Define sysplex policy
__________  NMC DEFINITIONS      Define NMC Data
__________  APPLICATION GROUPS   Select application groups for sysplex
__________  SYSPLEX DEFAULTS     Select sysplex resource defaults for group
__________  -------------------- -----LOCAL PAGE DATA SET POLICY--------------
__________  LOCAL PAGE DATA SET  Define local page data set recovery
__________  JOB DEFINITIONS      Define handling of jobs
__________  -------------------- -----LONG RUNNING ENQUEUE POLICY-------------
__________  JOB/ASID DEFINITIONS Define handling of long running jobs and ASID
__________  COMMAND  DEFINITIONS Define "hung" commands
__________  COMMAND  FLOODING    Define check intervals for command flooding
__________  RESOURCE DEFINITIONS Define long running enqueue resources
__________  RECOVERY OPTIONS     Define dump and miscellaneous options
__________  -------------------- --------------------------------------------
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

Figure 70. Policy Selection Panel for Sysplex Groups

The policy items for sysplex groups are the same as those for standard groups with the following additions:

SYSPLEX
This policy item allows you to describe certain resources associated with the sysplex. For more information, see “SYSPLEX Policy Item” on page 105.

NMC DEFINITIONS
This policy item allows you to make NMC definitions for the primary and secondary NMC Focal Point Domains. For more information, see “NMC DEFINITIONS Policy Item” on page 107.

APPLICATION GROUPS
This policy item displays a list of all Sysplex Application Groups, indicating which application groups are selected for this sysplex. For more information, see “APPLICATION GROUPS Policy Item” on page 109.

SYSPLEX DEFAULTS
When you select this policy item, it displays a list of all Sysplex Defaults entries in the enterprise. SELECTED in the **Status** column indicates the Sysplex Defaults entry that has been linked to the group. You can add a Sysplex Defaults set to or remove it from the group but only one set can be linked. For more information, see "Sysplex Defaults Policy Object" on page 307.

LOCAL PAGE DATA SET
With this policy item you can predefine local page data sets that are used in recovery situations to prevent auxiliary storage shortage outages. For more information, see "LOCAL PAGE DATA SET Policy Item" on page 109.

JOB DEFINITIONS
With this policy item you can define the jobs that can or cannot be canceled (KEEP/CANCEL), in case the job is responsible for the shortage condition. For more information, see "JOB DEFINITIONS Policy Item" on page 110.

JOB/ASID DEFINITIONS
With this policy item you can define the jobs and address space IDs that can or cannot be canceled. You can specify KEEP if the job must not be canceled, DUMP to dump the job after the wait time has expired, or NODUMP to suppress the dump before the job is canceled.

For more information, see "JOB/ASID DEFINITIONS Policy Item" on page 111.

COMMAND DEFINITIONS
With this policy item you can define which commands you want to monitor and define a maximum time that a command can be blocked by a job or another command before recovery action takes place.

You can also specify an action (KEEP, DUMP or NODUMP) to be taken against this command if it is found to be blocking other commands from executing.

For more information, see "COMMAND DEFINITIONS Policy Item" on page 112.

COMMAND FLOODING
With this policy item you can define individually the interval time for snapshot processing for each command class. When message IEE806A triggers the recovery, the first snapshot is taken after the interval has expired.

For more information, see "COMMAND FLOODING Policy Item" on page 113.

RESOURCE DEFINITIONS
With this policy item you can define DUMP and other miscellaneous recovery options, including the message delay time for message ING920I (issued when option KEEP is specified), and IEADMC*xx* symbols.

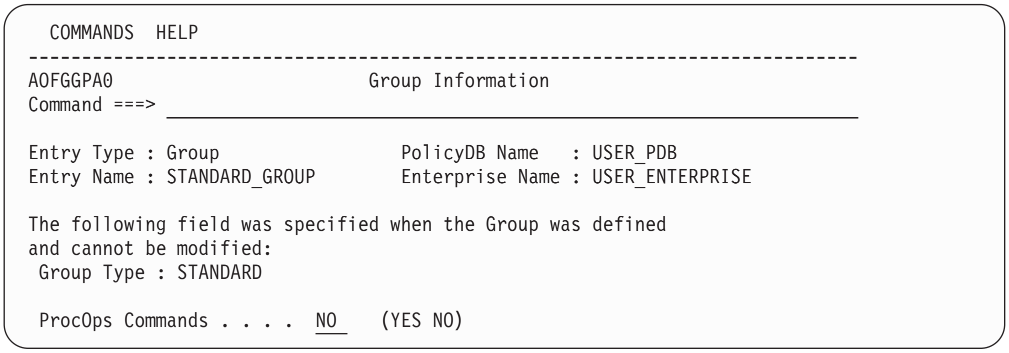
For more information, see "RESOURCE DEFINITIONS Policy Item" on page 114.

RECOVERY OPTIONS
With this policy item you can define DUMP and other miscellaneous recovery options including message delay time for message ING920I (issued when option KEEP is specified), and IEADMC*xx* symbols.

For more information, see “RECOVERY OPTIONS” on page 115.

GROUP INFO Policy Item

A panel as shown in Figure 71 is displayed if you select policy item GROUP INFO on the Policy Selection panel for standard groups. (If the group type was defined as sysplex, this is reflected in the panel.)



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGGPA0                        Group Information
Command ===> _________________________________________________________________

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : STANDARD_GROUP        Enterprise Name : USER_ENTERPRISE

The following field was specified when the Group was defined
and cannot be modified:
 Group Type : STANDARD

 ProcOps Commands . . . .  NO     (YES NO)
```

Figure 71. Group Information Panel for Standard Groups

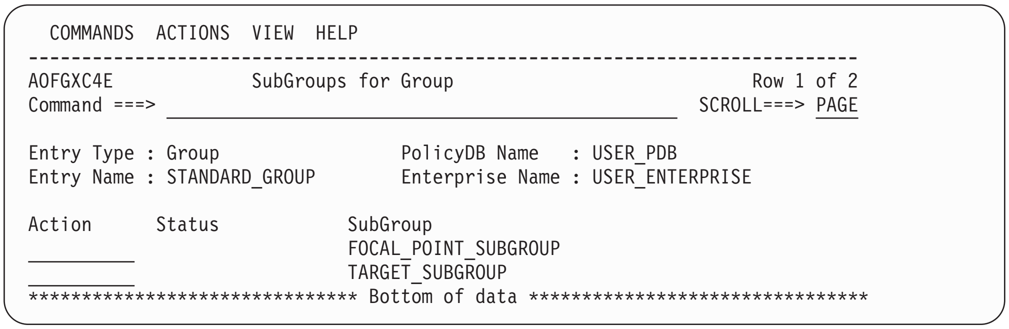
This panel displays the following information:

Group Type
This field cannot be changed; it was set when this group was created. For more details, see “Group Type” on page 100.

ProcOps Commands
For more details, see “ProcOps Commands ” on page 100.

SUBGROUPS Policy Item

A panel as shown in Figure 72 is displayed if you select policy item SUBGROUPS on the Policy Selection panel for standard groups. (If the group type was defined as sysplex, this is reflected in the panel.)



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E            SubGroups for Group                           Row 1 of 2
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : STANDARD_GROUP        Enterprise Name : USER_ENTERPRISE

Action      Status          SubGroup
__________                  FOCAL_POINT_SUBGROUP
__________                  TARGET_SUBGROUP
******************************* Bottom of data ********************************
```

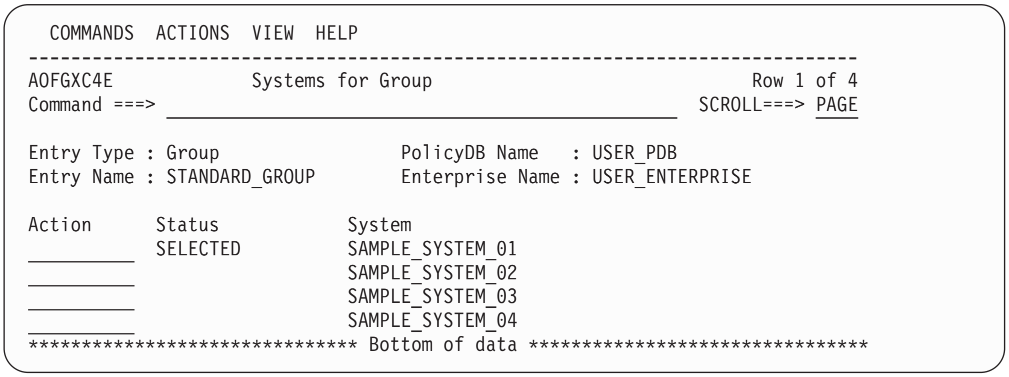
Figure 72. Subgroups Panel for Standard Groups

This panel allows you to manage the links between the current Group entry and the SubGroups entries. From this panel you can:

- Add a link between the entries with the ADD (A) command in the Action column.
- Remove a link between the entries with the REMOVE (M) command in the Action column.

SYSTEMS Policy Item

A panel as shown in Figure 73 is displayed if you select policy item SYSTEMS on the Policy Selection panel for standard groups. (If the group type was defined as sysplex, this is reflected in the panel.)



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E             Systems for Group                             Row 1 of 4
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : STANDARD_GROUP        Enterprise Name : USER_ENTERPRISE

Action      Status           System
__________  SELECTED         SAMPLE_SYSTEM_01
__________                   SAMPLE_SYSTEM_02
__________                   SAMPLE_SYSTEM_03
__________                   SAMPLE_SYSTEM_04
****************************** Bottom of data ********************************
```

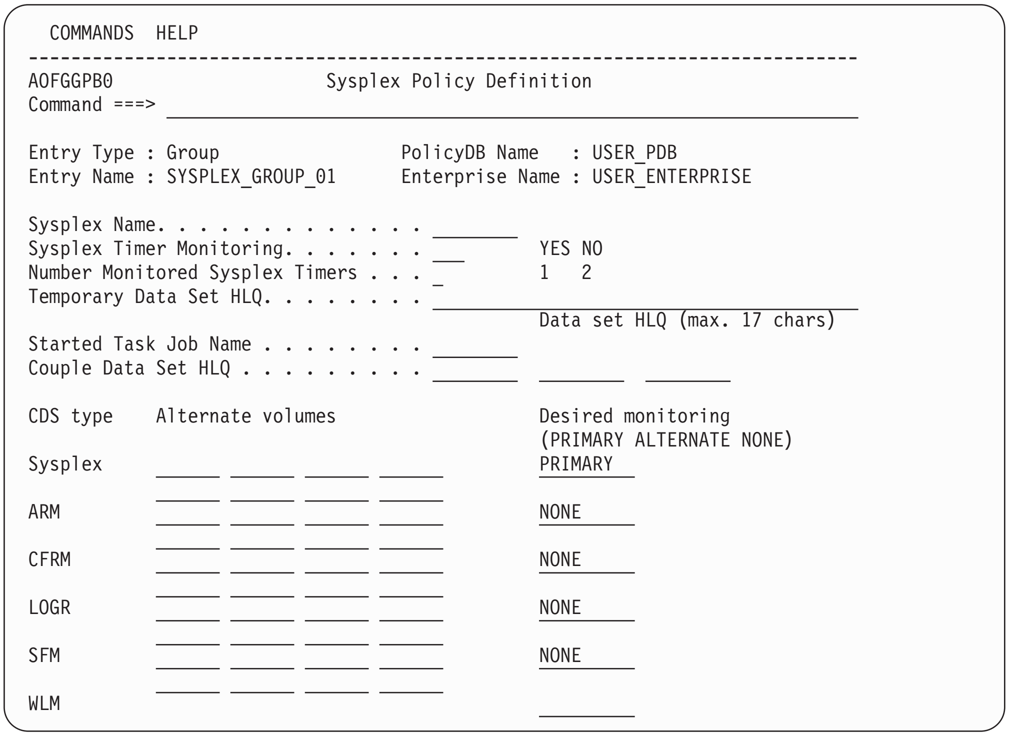
Figure 73. Systems Panel for Standard Groups

This panel allows you to manage the links between the current Group entry and the Systems entries. From this panel you can:

- Add a link between the entries with the ADD (A) command in the Action column.
- Remove a link between the entries with the REMOVE (M) command in the Action column.

SYSPLEX Policy Item

A panel as shown in Figure 74 on page 106 is displayed if you select policy item SYSPLEX on the Policy Selection panel for sysplexes.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGGPB0                    Sysplex Policy Definition
Command ===> _________________________________________________________________

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : SYSPLEX_GROUP_01      Enterprise Name : USER_ENTERPRISE

Sysplex Name. . . . . . . . . . . . . ________
Sysplex Timer Monitoring. . . . . . . ___       YES NO
Number Monitored Sysplex Timers . . . _         1   2
Temporary Data Set HLQ. . . . . . . . ________________________________________
                                                Data set HLQ (max. 17 chars)
Started Task Job Name . . . . . . . . ________
Couple Data Set HLQ . . . . . . . . . ________  ________  ________

CDS type    Alternate volumes                   Desired monitoring
                                                (PRIMARY ALTERNATE NONE)
Sysplex     ______ ______ ______ ______         PRIMARY__
            ______ ______ ______ ______
ARM         ______ ______ ______ ______         NONE_____
            ______ ______ ______ ______
CFRM        ______ ______ ______ ______         NONE_____
            ______ ______ ______ ______
LOGR        ______ ______ ______ ______         NONE_____
            ______ ______ ______ ______
SFM         ______ ______ ______ ______         NONE_____
            ______ ______ ______ ______
WLM                                             _________
```

Figure 74. Sysplex Policy Definition Panel for Sysplex Groups

The fields on the Sysplex Policy Definition panel are as follows:

Sysplex Name
Defines the name of the sysplex.

Sysplex Timer® Monitoring
If you have one or more sysplex timers in your sysplex and you want them monitored by the NMC workstation, enter YES and specify the number of monitored sysplex timers in the next field. Enter NO, if you do not want monitoring. This is the default.

Number Monitored Sysplex Timers
Enter ’2’ if you are using the sysplex timer with the Expanded Availability Option and you want to have two sysplex timers monitored by the NMC workstation. Enter ’1’, if you only want one sysplex timer monitored.

Temporary Data Set HLQ
Defines the high level qualifier to be used to assemble a data set name for allocating temporary data sets needed by programs running as started tasks. The qualifier can consist of up to 17 characters according to the z/OS data set naming rules (hlq1.hlq2.hlq3). Specify the name without quotes.

Started Task Job Name
Defines the job name to be used for programs running as started tasks. The name can consist of up to 8 characters according to the z/OS job naming rules.

Couple Data Set HLQ
Specify the high level qualifier to be used for the allocation of alternate couple data sets (CDS). The qualifier can consist of up to three parts according to the z/OS data set naming rules. When allocating an alternate CDS, this qualifier is appended with the type (for example, SYSPLEX) and CDS0*n*, where *n* is a sequence number.

CDS type - Alternate volumes - Desired monitoring
Specify the types of couple data sets to be used, and the monitoring to be done.

The types of supported couple data sets are:

Sysplex
Contains the policy and status for basic sysplex functions and points to the other couple data sets.

ARM Contains the policy for automatic restart management.

CFRM Contains the policy for coupling facility resource management.

LOGR Contains an inventory for the sysplex logger function.

SFM Contains the policy for sysplex failure management.

WLM Contains the policy for the workload manager.

In the column *Alternate volumes* you can enter up to 8 volume names for allocating alternate couple data sets for types Sysplex, ARM, CFRM, LOGR and SFM. For WLM, no alternate volumes can be specified. The names must be in accordance to the z/OS volume naming rules. In case of allocating alternate couple data sets, the allocation starts on the first specified volume and continues in the specified order.

The information you enter in column *Desired monitoring*, determines when the NMC workstation of SA z/OS displays the associated icon for a couple data set as satisfactory or unsatisfactory. For example, if you have a couple data set `QUAL1.ARM.CDS0`, which you want to be monitored as PRIMARY, but currently this data set is not available because the operator had to switch to an alternate ARM CDS, the icon for `QUAL1.ARM.CDS0` is marked as being unsatisfactory (colored red).

You specify the monitoring that is to be done for each type of couple data set as follows:

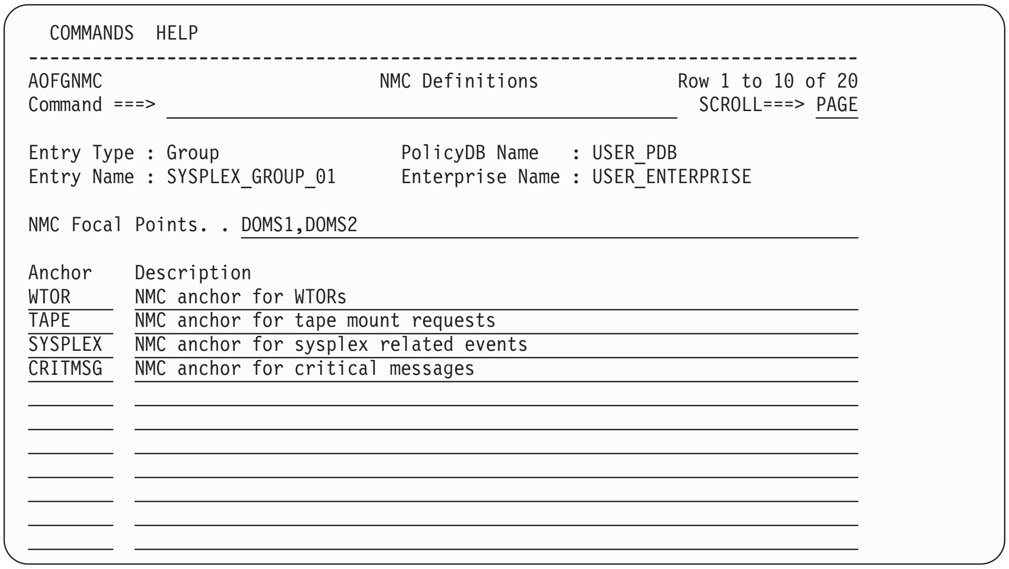
ALTERNATE
Indicates that you want to monitor both the primary and the alternate data set. In this case, when either the primary or the alternate fails, the related icon will turn red.

PRIMARY
Indicates that you want to monitor only the primary couple data set. In this case, there is only one icon (for the primary CDS) on the NMC workstation and the operator will only see when the primary itself is unavailable, no matter whether an alternate CDS is available to take over.

NONE
Indicates that none of the couple data sets are monitored. This is the default for all types of CDSs, except for the sysplex CDS. The default for the sysplex CDS is PRIMARY.

NMC DEFINITIONS Policy Item

A panel as shown in Figure 75 on page 108 is displayed if you select policy item NMC DEFINITIONS on the Policy Selection panel for sysplexes.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGNMC                          NMC Definitions            Row 1 to 10 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : SYSPLEX_GROUP_01      Enterprise Name : USER_ENTERPRISE

NMC Focal Points. . DOMS1,DOMS2___________________________________________

Anchor    Description
WTOR      NMC anchor for WTORs
TAPE      NMC anchor for tape mount requests
SYSPLEX   NMC anchor for sysplex related events
CRITMSG   NMC anchor for critical messages
________  ____________________________________________________________________
________  ____________________________________________________________________
________  ____________________________________________________________________
________  ____________________________________________________________________
________  ____________________________________________________________________
________  ____________________________________________________________________
```

Figure 75. NMC Definitions Panel for Sysplex Groups

The NMC Definitions panel displays the following information:

NMC Focal Points

Defines a list of NetView domain IDs that act as focal points. The domain IDs should be in the form ’netid.domain’ or ’domain’ and be separated by commas. You can include system symbols and system automation symbols.

Anchor

Allows you to specify anchors that are special objects in RODM and displayed on the topology client of the SA z/OS Graphic Interface. They are used to display and group minor resources that are derived from events such as WTORs and tape mount requests.

Anchors are special objects that exist in RODM. They are displayed as icons on the NMC topology client. They are not automated by the automation manager. They play the role of major resources and serve these purposes:

- Anchors are used to display and group minor resources that are derived from events such as WTORS and tape mount requests.
- Lost&Found anchors are used to display minor resources that either the specified major resource or sysplex anchor does not exist for in RODM.
- Couple data sets (CDSs), coupling facilities (CFs) and external timer references (ETRs) are displayed as minor resources and are attached to the sysplex anchor.
- It is possible to attach a minor resource to a user-defined anchor with the INGPOST command. The major resource specified to INGPOST can be a regular resource (for example, APL, APG, SYS, etc.) or an anchor.
- Heartbeats (minor resources) are attached to a heartbeat anchor.

Specify the name of an anchor on this panel for these cases:

- To enable the display of WTORS and tape mount requests in the NMC topology client, the anchor names that must be specified are `WTOR` and `TAPE` respectively
- To define a sysplex anchor for ’homeless’ minor resources, or to display CDSs, CFs, and ETRs, the name of the sysplex anchor is `SYSPLEX`

- To display minor resources that are created with INGPOST. The name used for the `resource=` parameter should be the name of the anchor that is specified here. For example, with `resource=ANCHOR/MJR/CRITMSG` the name of the anchor is `CRITMSG`

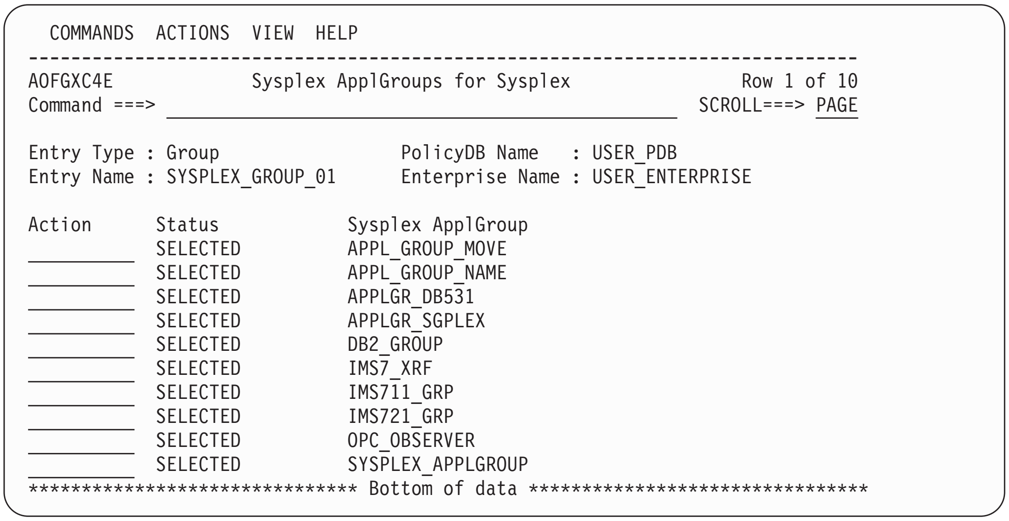
Do *not* specify an anchor on this panel for:

- Heartbeats. The anchor is automatically created by SA z/OS.
- Lost&Found anchors.

Description
Allows you to specify a description for the anchor. The value is only shown in the SA z/OS customization dialog panel. It is not displayed in the SA z/OS Graphic Interface.

APPLICATION GROUPS Policy Item

A panel as shown in Figure 76 is displayed if you select policy item APPLICATION GROUPS on the Policy Selection panel for sysplex groups.



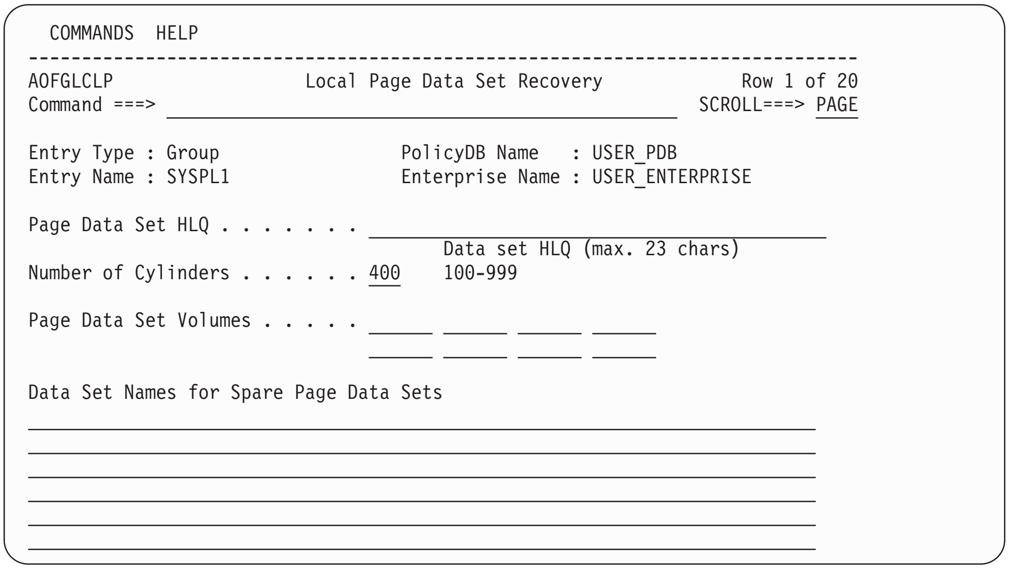
```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E             Sysplex ApplGroups for Sysplex               Row 1 of 10
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : SYSPLEX_GROUP_01      Enterprise Name : USER_ENTERPRISE

Action      Status           Sysplex ApplGroup
__________  SELECTED         APPL_GROUP_MOVE
__________  SELECTED         APPL_GROUP_NAME
__________  SELECTED         APPLGR_DB531
__________  SELECTED         APPLGR_SGPLEX
__________  SELECTED         DB2_GROUP
__________  SELECTED         IMS7_XRF
__________  SELECTED         IMS711_GRP
__________  SELECTED         IMS721_GRP
__________  SELECTED         OPC_OBSERVER
__________  SELECTED         SYSPLEX_APPLGROUP
******************************* Bottom of data ********************************
```

Figure 76. Group Information Panel for Standard Groups

This panel allows you to manage the links between the current Sysplex entry and the Sysplex ApplGroups entries.

LOCAL PAGE DATA SET Policy Item

With the automation of local page data sets, SA z/OS prevents auxiliary storage shortage outages, by dynamically allocating spare local page data sets when needed.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGLCLP                 Local Page Data Set Recovery            Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : SYSPL1                Enterprise Name : USER_ENTERPRISE

Page Data Set HLQ . . . . . . . ___________________________________________
                                       Data set HLQ (max. 23 chars)
Number of Cylinders . . . . . . 400    100-999

Page Data Set Volumes . . . . . ______ ______ ______ ______
                                ______ ______ ______ ______

Data Set Names for Spare Page Data Sets
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
__________________________________________________________________________
```

Figure 77. Local Page Data Set Recovery Panel

Fill in the following fields:

Page Data Set HLQ
Defines a high level qualifier which is used to assemble a data set name for creating and allocating a page data. The qualifier can consist of up to 23 characters and must follow the naming conventions of MVS data sets (hlq.hlq2.hlq3). The high level qualifier must point to the master catalog and must not be SMS managed.

Number of Cylinders
Defines the maximum number of cylinders (100 to 999) used for the dynamic allocation of a local page data set. The default value is 400. Note that on a 3390 DASD, 100 cylinders is adequate to 70 MB and the formatting process takes approximately 20 seconds for this amount of space.

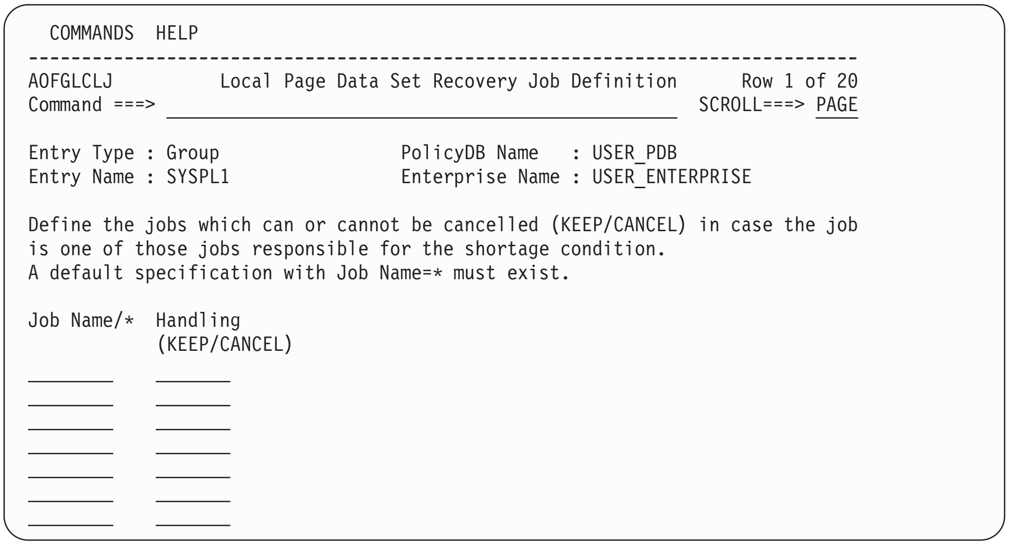
Page Data Set Volumes
Defines a list of volume IDs which are eligible when the automation is creating and allocating a new page data set. The volumes must be shared by all systems in the sysplex.

Data Set Names for Spare Page Data Sets
Defines the pre-formatted spare page data set to be used in the recovery situation. Specify the names without quotes. You may include system symbols and system automation symbols (see “Assigning System Automation Symbols (AOCCLONE)” on page 27). The data sets must be allocated on a volume shared by all systems in the sysplex and must be catalogued in the master catalog. In case the systems in the sysplex do not share the master catalog, the page data set must be re-catalogued on each system except the one which created the data set using the following IDCAMS statement:

```
DEF PAGESPACE(NAME(lpdsn) VOLUME(volume) RECATALOG)
```

JOB DEFINITIONS Policy Item

You can customize the automation to cancel or keep jobs that are blocking a resource.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGLCLJ         Local Page Data Set Recovery Job Definition      Row 1 of 20
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Group                PolicyDB Name   : USER_PDB
Entry Name : SYSPL1               Enterprise Name : USER_ENTERPRISE

Define the jobs which can or cannot be cancelled (KEEP/CANCEL) in case the job
is one of those jobs responsible for the shortage condition.
A default specification with Job Name=* must exist.

Job Name/*  Handling
            (KEEP/CANCEL)

________    _______
________    _______
________    _______
________    _______
________    _______
________    _______
________    _______
```

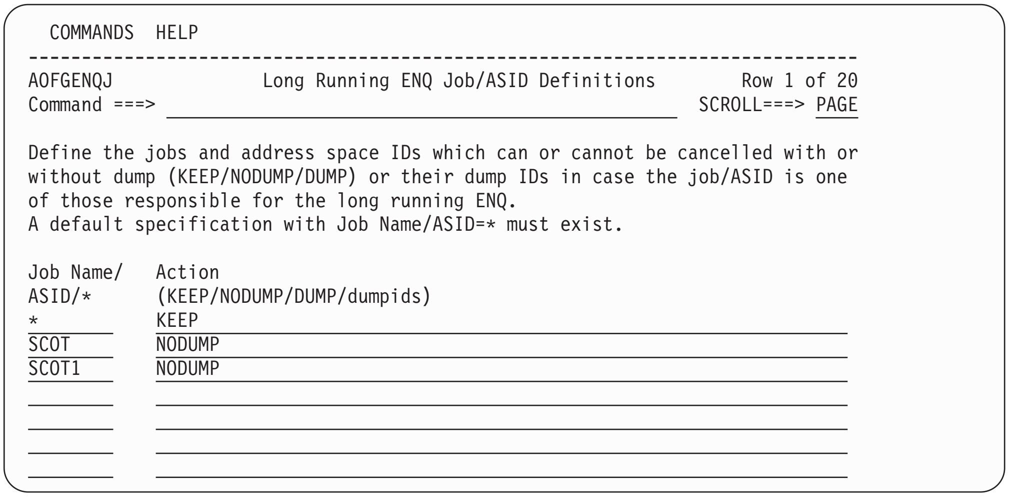
Figure 78. Job Definitions for Local Page Data Set Recovery Panel

Job Name
: Defines the jobs which can or cannot be canceled in case they are one of those jobs responsible for the shortage condition.

Handling
: Specify KEEP if the job cannot be canceled. Otherwise specify CANCEL. You can also specify a wildcard character (*). You must define a default with Job Name=* and either KEEP if all jobs not explicitly defined cannot be canceled, or CANCEL if all jobs can be canceled.

JOB/ASID DEFINITIONS Policy Item

Entries specified here apply to long-running ENQ, hung-command and command-flooding recovery. Specify the names of Jobs or ASIDs that require special treatment (KEEP, DUMP or NODUMP) when a TCB in the Job or ASID is found to be causing a long-running ENQ lockout, a hung-command situation or command flooding.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGENQJ             Long Running ENQ Job/ASID Definitions        Row 1 of 20
Command ===> ________________________________________________ SCROLL===> PAGE

Define the jobs and address space IDs which can or cannot be cancelled with or
without dump (KEEP/NODUMP/DUMP) or their dump IDs in case the job/ASID is one
of those responsible for the long running ENQ.
A default specification with Job Name/ASID=* must exist.

Job Name/   Action
ASID/*      (KEEP/NODUMP/DUMP/dumpids)
*           KEEP
SCOT        NODUMP
SCOT1       NODUMP
________    _________________________________________________________________
________    _________________________________________________________________
________    _________________________________________________________________
________    _________________________________________________________________
```

Figure 79. JOB/ASID Definitions Panel

Fill in the following fields:

Job Name/ASID
Defines the jobs and address space IDs that can or cannot be canceled in case they are one of those that are responsible for a long running ENQ or hung command.

Action
Specify:

- KEEP if the job or ASID cannot be canceled. Message ING920I will be issued at user-defined intervals as long as the ENQ or hung command situation persists (see "Message Delay field" on page 115).
- NODUMP if the job or ASID can be canceled without a dump.
- DUMP if the job or ASID can be canceled with a dump. DUMP options are specified under RECOVERY OPTIONS (see "RECOVERY OPTIONS" on page 115).

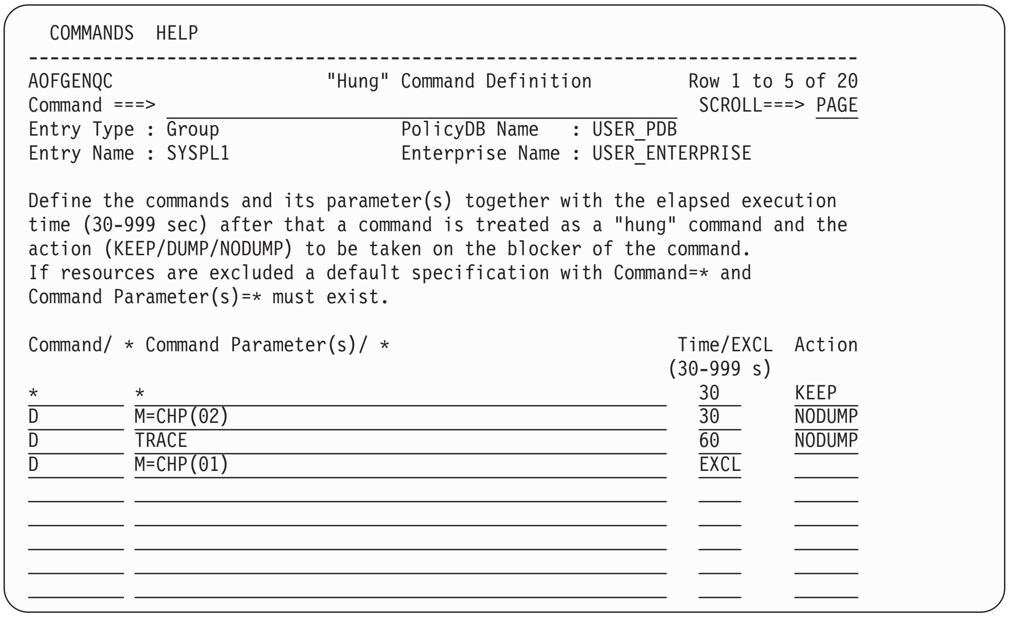
Specification of a trailing wildcard (*) is allowed.

For long-running ENQ recovery you must define a default entry with `Job name/ASID=*` together with the action that will be appropriate for all jobs and ASIDs that are not explicitly defined. For hung-command recovery this default entry is ignored.

COMMAND DEFINITIONS Policy Item

With this policy item you can define which commands you want to monitor and define a maximum time that a command can be blocked by a job or another command before recovery action takes place.

You can also define an action (KEEP, DUMP or NODUMP) to be taken against this command if it is found to be blocking other commands from executing.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGENQC                    "Hung" Command Definition         Row 1 to 5 of 20
Command ===> ________________________________________________ SCROLL===> PAGE
Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : SYSPL1                Enterprise Name : USER_ENTERPRISE

Define the commands and its parameter(s) together with the elapsed execution
time (30-999 sec) after that a command is treated as a "hung" command and the
action (KEEP/DUMP/NODUMP) to be taken on the blocker of the command.
If resources are excluded a default specification with Command=* and
Command Parameter(s)=* must exist.

Command/ * Command Parameter(s)/ *                            Time/EXCL  Action
                                                             (30-999 s)
*          *                                                    30       KEEP
D          M=CHP(02)                                            30       NODUMP
D          TRACE                                                60       NODUMP
D          M=CHP(01)                                            EXCL
```

Figure 80. "Hung" Command Definitions Panel

Fill in the following fields:

Command
Specify the name of a z/OS system command that is to be checked whether it is a long running command (for example, the z/OS command SET).

Note: The MODIFY command (F) cannot be processed by hung command recovery because it does not enqueue on any system resource. There is no need to automate the MODIFY command.

Command Parameter(s)
Specify the parameter (or parameters) of a z/OS command that is to be checked whether it is a long running command (for example, PROG as the parameter of the z/OS command SET).

Time Defines the elapsed execution time after which a command is treated as a long running command. Valid value is 30-999 seconds.

If the command should be excluded from long running command checking, specify `EXCL`.

Action
Defines the action to be taken (KEEP, DUMP or NODUMP) when the specified command blocks other commands from executing.

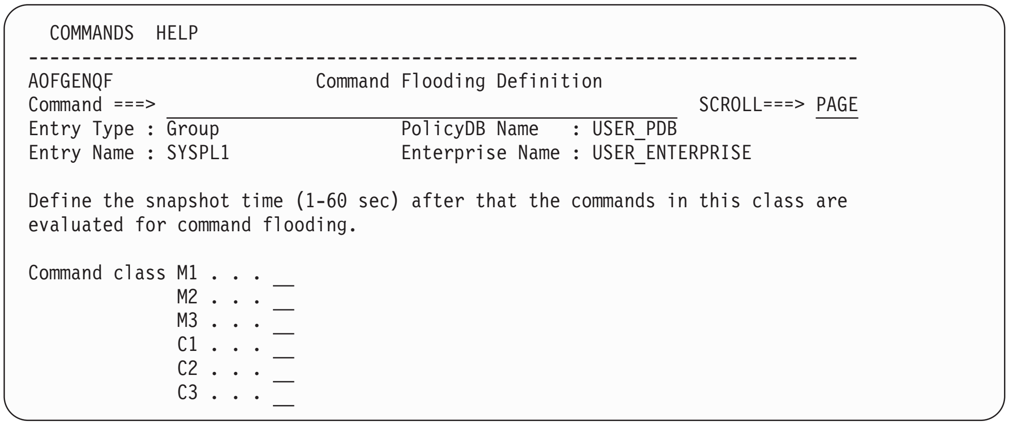
The action specified here will be overridden by the action that is specified in the Jobs/ASID policy item for the job or ASID that issued the blocking command (often *MASTER* or CONSOLE).

For example, if a TSO user named FRED issued a D TRACE command that hung and blocked other commands from executing, and there is a Hung Command Definition of DISPLAY TRACE,30,NODUMP then, after 30 seconds the following would happen for different Job/ASID Definitions:

Job/ASID Definition	Result
FRED,KEEP	FRED's D TRACE command would not be canceled but message ING920I would be issued because *specific* Job/ASID definitions take precedence over COMMAND DEFINITIONs.
–	The D TRACE command would be canceled without a dump.
*,KEEP	The D TRACE command would be canceled without a dump because COMMAND DEFINITIONs take precedence over *default* Job/ASID definitions.

COMMAND FLOODING Policy Item

You can define individually the interval time for snapshot processing for each command class. When message IEE806A triggers the recovery, the first snapshot is taken after the interval has expired.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGENQF                   Command Flooding Definition
Command ===> ________________________________________________ SCROLL===> PAGE
Entry Type : Group                PolicyDB Name   : USER_PDB
Entry Name : SYSPL1               Enterprise Name : USER_ENTERPRISE

Define the snapshot time (1-60 sec) after that the commands in this class are
evaluated for command flooding.

Command class M1 . . . __
              M2 . . . __
              M3 . . . __
              C1 . . . __
              C2 . . . __
              C3 . . . __
```

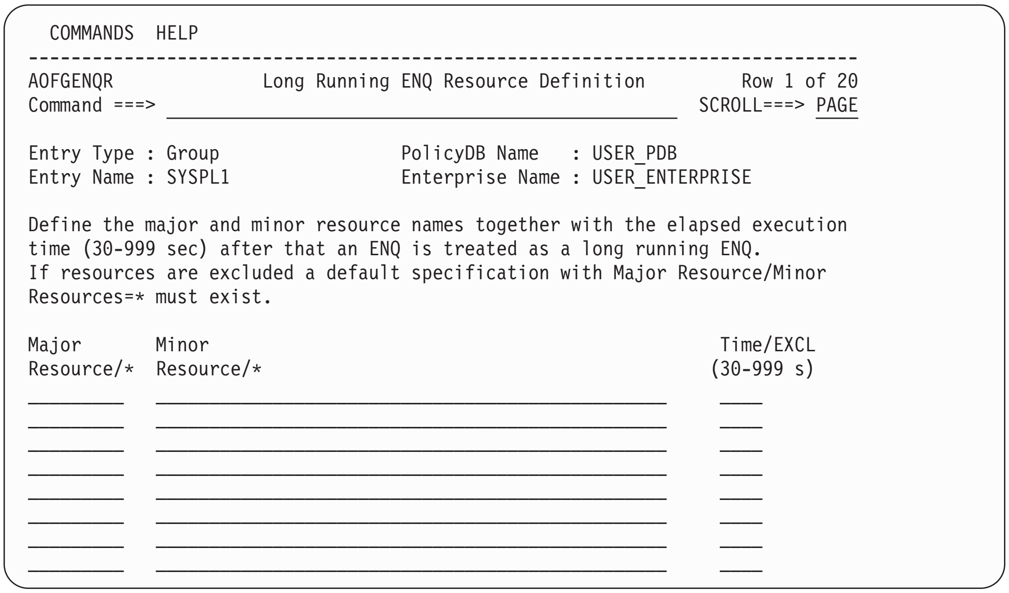
Figure 81. Command Class Definitions Panel

Fill in the following field:

Snapshot time
Defines the time the recovery routine should wait before it takes the next snapshot of the commands that are being executed or waiting in the command class. The time should be long enough to allow a task that issues a set of commands to finish before the first snapshot is taken. Valid value is 1-60 seconds. A default value of 3 seconds is used if nothing is specified.

RESOURCE DEFINITIONS Policy Item

You can specify which resources you want to monitor and define a value for the maximum time that a job can lock a resource while other jobs are waiting for the resource. If the amount of time is exceeded, the recovery takes place.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGENQR              Long Running ENQ Resource Definition         Row 1 of 20
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Group                PolicyDB Name   : USER_PDB
Entry Name : SYSPL1               Enterprise Name : USER_ENTERPRISE

Define the major and minor resource names together with the elapsed execution
time (30-999 sec) after that an ENQ is treated as a long running ENQ.
If resources are excluded a default specification with Major Resource/Minor
Resources=* must exist.

Major       Minor                                                 Time/EXCL
Resource/*  Resource/*                                           (30-999 s)
_________   ________________________________________________      ____
_________   ________________________________________________      ____
_________   ________________________________________________      ____
_________   ________________________________________________      ____
_________   ________________________________________________      ____
_________   ________________________________________________      ____
_________   ________________________________________________      ____
_________   ________________________________________________      ____
```

Figure 82. Long Running ENQ Resource Definition Panel

Fill in the following fields:

Major Resource Name
Defines the major resource name of the resource to be checked for long running enqueues.

Minor Resource Name
Defines the minor resource name of the resource to be checked for long running enqueues.

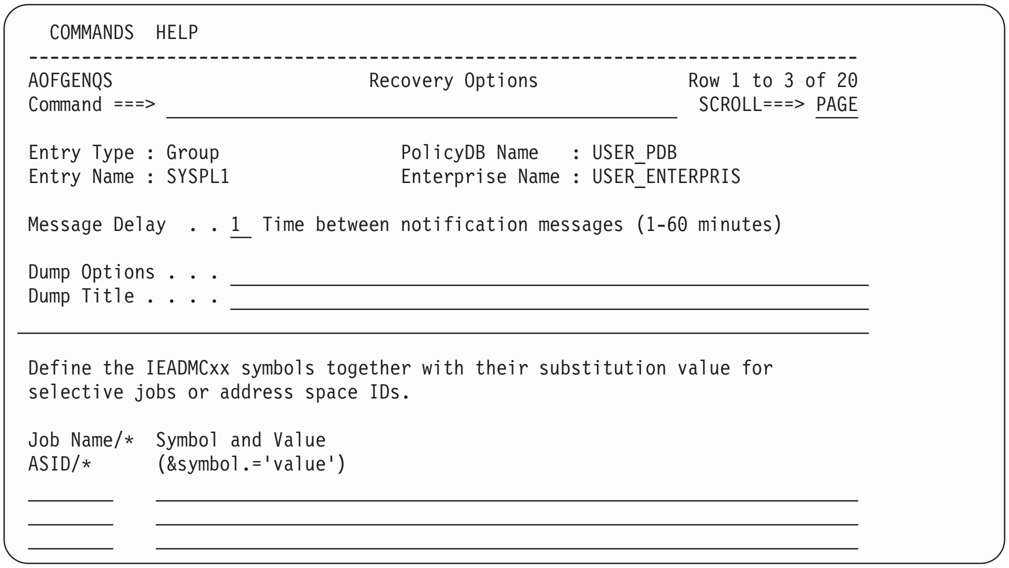
Time/EXCL
Defines the elapsed execution time after which an enqueue is treated as a long-running ENQ. Valid value is 30-999 seconds. If the resource should be excluded from long-running enqueue checking, specify `EXCL`.

RECOVERY OPTIONS

If you want to automate long-running ENQs, ″hung″ commands, or command flooding, you can specify dump options, a title for each dump being taken, and job names or address space IDs of the jobs that are kept or canceled.

Note that the dump options do not apply to command flooding because the recovery routine of the job being canceled can suppress the dump.

With this policy item you can define DUMP and other miscellaneous recovery options including message delay time for message ING920I (issued when option KEEP is specified), and IEADMC*xx* symbols.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGENQS                        Recovery Options             Row 1 to 3 of 20
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Group                 PolicyDB Name   : USER_PDB
Entry Name : SYSPL1                Enterprise Name : USER_ENTERPRIS

Message Delay  . . 1_ Time between notification messages (1-60 minutes)

Dump Options . . . ____________________________________________________________
Dump Title . . . . ____________________________________________________________
_______________________________________________________________________________

Define the IEADMCxx symbols together with their substitution value for
selective jobs or address space IDs.

Job Name/*  Symbol and Value
ASID/*      (&symbol.='value')
________    __________________________________________________________________
________    __________________________________________________________________
________    __________________________________________________________________
```

Figure 83. Recovery Options Panel

Fill in the following fields:

Message Delay
Defines the number of minutes that automation waits before it reissues the message indicating that a job or a command is still blocking other jobs or commands but that the blocker must not be canceled.

Note: Leaving the field blank results in a delay of 15 minutes.

Dump Options
Defines the DUMP option that is used for the SDATA parameter on the dump command. The following options are allowed:

ALLNUC
COUPLE
CSA
LPA
LSQA
NOPSA
NOSUM
NUC
PSA
SERVERS
SQA
SUM
TRT
WLM
XESDATA

GRSQ NOSQA RGN SWA

Note: This value applies only to the jobs or address space IDs that are defined with DUMP in the following list.

Dump Title
Defines the title of each dump being taken with the default dump options.

Job Name/ASID
Defines the jobs and address space IDs together with their symbols and substitution value for use with the dump command parmlib members IEADMC*xx*.

Specification of a trailing wildcard (*) is allowed.

Symbol and Value
Defines the symbol names and their substitution value in the format `&symbol='value'`.

Valid characters for the symbol are A – Z, 0 – 9, @, # and $. The value must be enclosed in single quotes and its length must not be longer than the symbol name.

SubGroup Policy Object

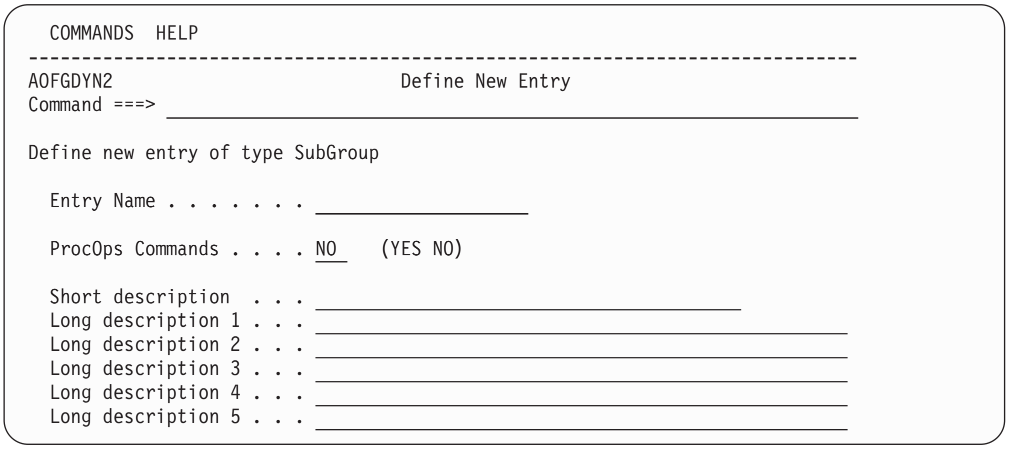
A subgroup is an intermediary grouping of the systems within a group. You define a subgroup by selecting the member systems and assigning a name to the subgroup. You do not have to define subgroups, but it may be useful to do so for a large enterprise. For example, a sysplex defined as a group may have subgroups consisting of systems running on one or more S/390® microprocessors within the sysplex so they can be managed separately from systems running on the other CPC types. Another option is to form subgroups based on system type, for example, all VM systems in one subgroup.

A system in a sysplex can belong to more than one subgroup within the sysplex group.

Creating a New SubGroup

How to create new policy objects in general, is described in “How to Create New Policy Objects” on page 37.

Figure 84 on page 117 is an example of the Define New Entry panel for subgroups.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN2                           Define New Entry
Command ===> _________________________________________________________________

Define new entry of type SubGroup

  Entry Name . . . . . . . ____________________

  ProcOps Commands . . . . NO_   (YES NO)

  Short description  . . . ________________________________________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 84. Define New Entry Panel for a Subgroup

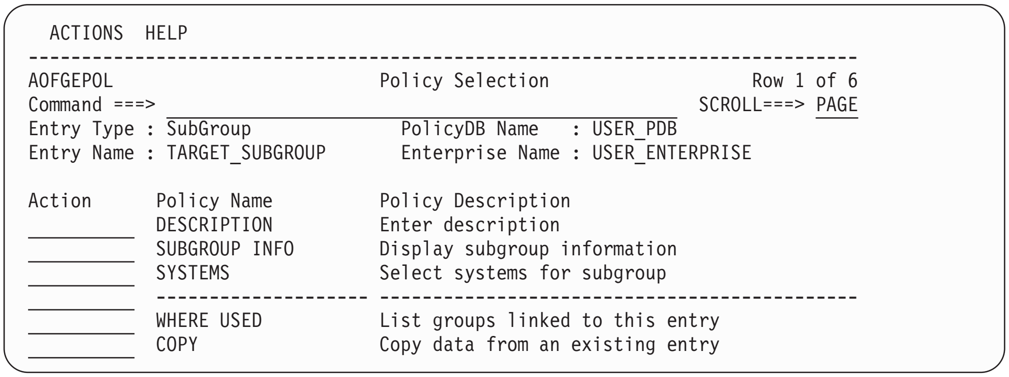
The unique field on this panel is:

ProcOps Commands
Indicates whether processor operations commands issued against this subgroup are to be accepted for processing. The default is NO.

Note:
If a group with which this subgroup is associated is enabled for processor operations commands, a processor operations command directed to that group will affect all systems in the group even if this subgroup is *not* enabled for processor operations commands.

Policy Items for SubGroups

Figure 85 shows the policy items that are displayed when a subgroup is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                         Policy Selection                  Row 1 of 6
Command ===> __________________________________________________ SCROLL===> PAGE
Entry Type : SubGroup              PolicyDB Name   : USER_PDB
Entry Name : TARGET_SUBGROUP       Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  SUBGROUP INFO        Display subgroup information
__________  SYSTEMS              Select systems for subgroup
__________  -------------------- --------------------------------------------
__________  WHERE USED           List groups linked to this entry
__________  COPY                 Copy data from an existing entry
```

Figure 85. Policy Selection Panel for Subgroups

The unique policy items for subgroups are:

SUBGROUP INFO
When you select this policy item, information about subgroups is displayed for verification and updating.

SYSTEMS
When you select this policy item, a list of all systems in the SA z/OS

enterprise is displayed, indicating those systems that are related to this subgroup. You can add systems to the subgroup, or remove systems from the subgroup.

System Policy Object

A system is any host system in the SA z/OS enterprise. The operating systems can be either MVS or non-MVS. For system operations, the system must be MVS. For processor operations functions, the system can be MVS, VM, VSE, LINUX, or CF (coupling facility). A system definition includes:

- The operating system type
- Processor operations information
- Details about which IBM automation products are operating on the system
- Member application groups
- NetView and z/OS-related information
- Automation policy for the system (optional)

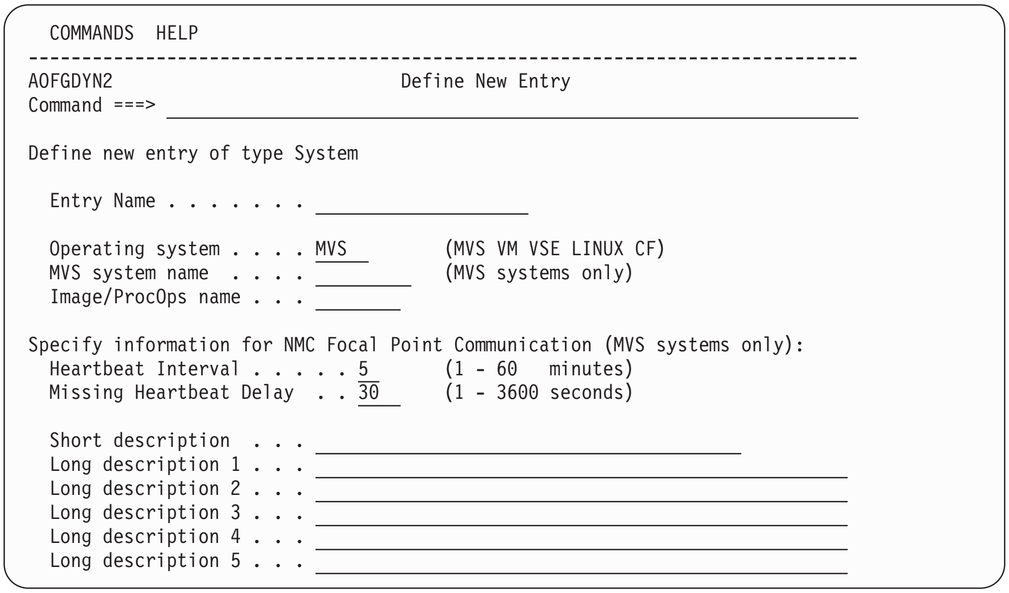
The information you supply here will determine what automation policy set objects are included in your system and how your system is automated.

Creating a New System

How to create new policy objects in general, is described in "How to Create New Policy Objects" on page 37.

Figure 86 shows the panel that is displayed when you create a new system from the Entry Name Selection panel for systems.

If you want to monitor your newly created system as a standalone system on the NMC workstation, you need to specify an entry for this sysplex in the INGTOPOF file (see *IBM Tivoli System Automation for z/OS Planning and Installation*).



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN2                           Define New Entry
Command ===> _________________________________________________________________

Define new entry of type System

  Entry Name . . . . . . . ____________________

  Operating system . . . . MVS_____    (MVS VM VSE LINUX CF)
  MVS system name  . . . . _________   (MVS systems only)
  Image/ProcOps name . . . ________

Specify information for NMC Focal Point Communication (MVS systems only):
  Heartbeat Interval . . . . . 5__     (1 - 60   minutes)
  Missing Heartbeat Delay  . . 30__    (1 - 3600 seconds)

  Short description  . . . ________________________________________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 86. Define New Entry Panel for a System

The fields are:

Name This required name gives a unique name to the system you are defining.

Valid characters are any alphabetic or numeric character or any of @, #, $, ?, or _ (underscore), with the first character being alphabetic, numeric, or @, #, or $. The entry name must be unique among the names of all groups, subgroups, systems, or processors, defined in the current policy database. It must also be unique among processor operations names, with the exception that this system's processor operations name may match this system's entry name.

Operating system

A required identifier of the operating system to be run on the system. The allowed values are MVS, VM, VSE, LINUX, or CF. The supported default is MVS.

CF indicates the coupling facility control code. LINUX identifies the Linux for zSeries operating system.

MVS SYSNAME

Required for MVS systems, this value *must match* the system name (SYSID) as specified in the IEASYS*xx* member of SYS1.PARMLIB. Valid characters are any alphabetic or numeric character or any of @, #, or $.

Note:

The first three characters of MVS SYSNAME must not be `AOF` because these are reserved for internal SA z/OS processing.

To support an alternate configuration of a system, special notation with a trailing 'plus-sign' is allowed. This trailing character will be removed when the automation files are built. If you are going to enable automation for IXC102A/IXC402D the MVS SYSNAME must not start with a numeric character. For more information see *IBM Tivoli System Automation for z/OS Customizing and Programming*.

Image/ProcOps name

Specifies the name of the system for hardware related functions, for example, processor operations functions.

Like the system's Entry name, this name must be unique among the names of all groups, subgroups, systems, processors and image names defined in the current policy database. For MVS systems the name is defaulted to the MVS SYSNAME but can be changed.

Valid characters are any alphabetic or numeric character or any of @, # or $. The name may start with a numeric character except for systems of type CF.

Heartbeat Interval

This specifies the time interval at which the SA z/OS system operations functions sends a generic alert from this system to the NMC focal point system. The SA z/OS system operations functions on the NMC focal point system use heartbeat alerts to verify the status forwarding path from each sysplex or single remote system. For more information see "The SA z/OS Heartbeat Function" on page 120. The heartbeat interval is a required field in the range 1 to 60 minutes that defaults to 5 minutes. Heartbeat information is relevant for z/OS systems only.

The smaller the heartbeat interval, the faster SA z/OS can detect a status forwarding path failure. For example, if you define an interval of two minutes then, on average, SA z/OS will take one minute, plus the missing heartbeat delay time, to detect a failure in the status forwarding path.

Missing Heartbeat Delay

This specifies the time that the SA z/OS system operations functions on the NMC focal point system will wait after a heartbeat was expected. If this time expires without receiving the heartbeat, the system operations functions will begin missing heartbeat processing for the remote system. The field is provided as a way to buffer against temporary network delays. For more information see “The SA z/OS Heartbeat Function.” The missing heartbeat delay is a required field in the range 1 to 3600 seconds that defaults to 30 seconds. Heartbeat information is relevant for z/OS systems only.

The smaller the missing heartbeat delay, the more sensitive SA z/OS is to temporary status forwarding path delays. The longer the delay, the slower SA z/OS will be to react to a missing heartbeat. Again, you will have to balance these two considerations.

Consideration must be given to the number of systems and subsystems being monitored in your enterprise. The larger the enterprise being monitored, the larger the delay should be. This is to ensure that at resync time with high status forwarding to the focal point, any delays due to this do not cause unnecessary missing heartbeat processing.

The SA z/OS Heartbeat Function

The heartbeat function validates the path between remote systems and the focal point system. This path is known as the status forwarding path.

SA z/OS validates the status forwarding path by sending an alert, called the heartbeat alert, from each single remote system or from a designated system in the sysplex at a regular interval.

If the heartbeat is missing from the designated sysplex system, the SA z/OS topology manager tries to connect to another system in the sysplex. Only if all systems are not responding, SA z/OS invokes the missing heartbeat processing.

SA z/OS on the NMC focal point system expects heartbeats from each remote system at the interval defined for each system, but it expects a heartbeat only from one system in the sysplex. This interval is known as the heartbeat interval. Receiving a heartbeat on the NMC focal point system indicates that the status forwarding path from the remote system is up, and hence status for resources associated with that system can be considered to be current.

If SA z/OS on the NMC focal point system does not detect an expected heartbeat, it waits to allow for temporary network slowdowns, or any other temporary problems in the status forwarding path that may have delayed the heartbeat. The wait period is known as the missing heartbeat delay. When the missing heartbeat delay time expires, (or for a sysplex, when the missing heartbeat delay time expires for all systems in the sysplex) SA z/OS invokes its missing heartbeat processing for the remote system or the sysplex and changes the status of all associated resources to Unknown because SA z/OS is no longer able to receive status updates for those resources.

When setting the missing heartbeat delay period, consideration must be given to the number of sysplexes or single systems, and subsystems being monitored in your enterprise. The larger the enterprise being monitored, the larger the delay should be.

When SA z/OS detects that heartbeats from a remote system have resumed, it resynchronizes the NMC workstation status displayed for resources associated with the system by requesting that the remote system should resend all status information.

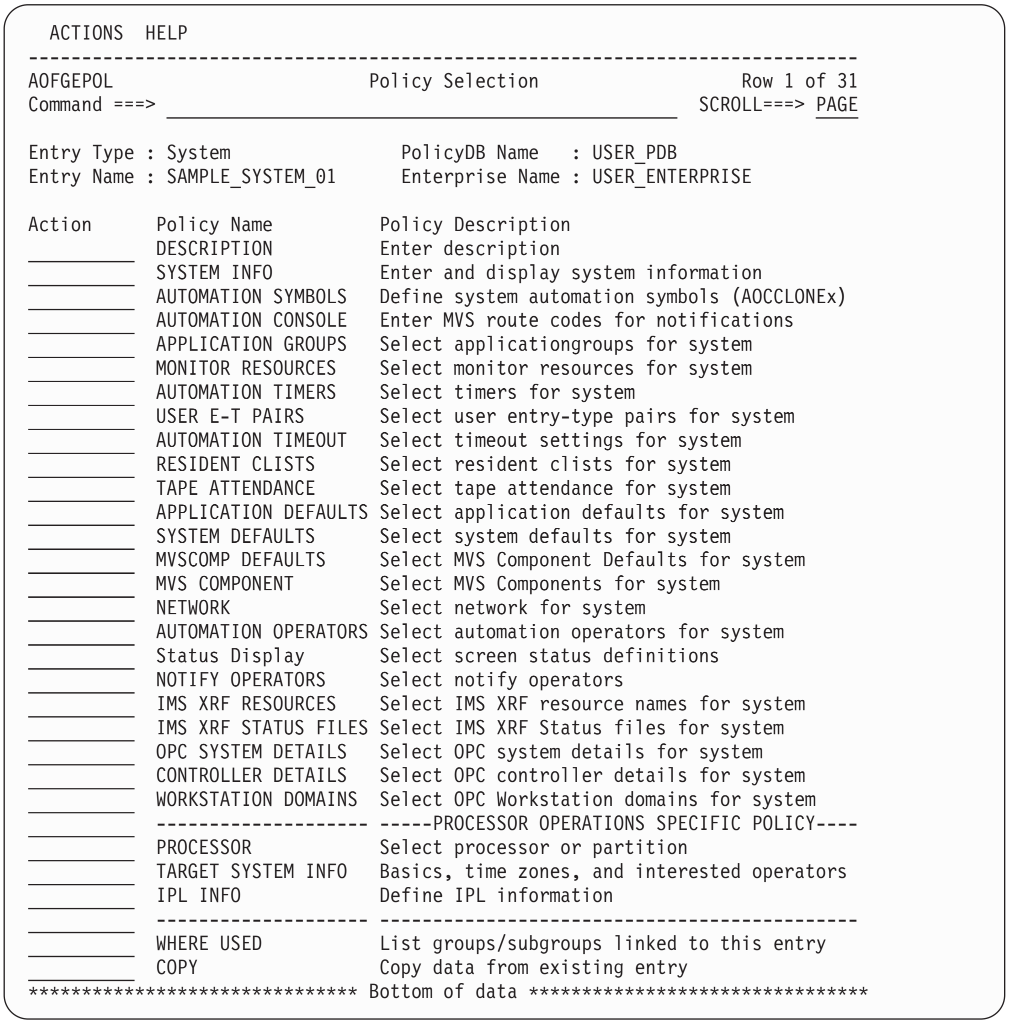
You can define both the heartbeat interval and the missing heartbeat delay for each MVS system in your enterprise.

Policy Items for Systems

Figure 87 on page 122 shows the z/OS policy items that are displayed when an MVS system is selected from the Entry Name Selection panel.

Notes:

- Not all policies are displayed on one screen. Scroll down to see the second part of this panel.
- Several of the policy items available from this panel allow you to select and manage the links between the system and a variety of entry types. The panels associated with these policy items are all very similar and straightforward to use and are not covered in this section.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                 Row 1 of 31
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : System                PolicyDB Name   : USER_PDB
Entry Name : SAMPLE_SYSTEM_01      Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  SYSTEM INFO          Enter and display system information
__________  AUTOMATION SYMBOLS   Define system automation symbols (AOCCLONEx)
__________  AUTOMATION CONSOLE   Enter MVS route codes for notifications
__________  APPLICATION GROUPS   Select applicationgroups for system
__________  MONITOR RESOURCES    Select monitor resources for system
__________  AUTOMATION TIMERS    Select timers for system
__________  USER E-T PAIRS       Select user entry-type pairs for system
__________  AUTOMATION TIMEOUT   Select timeout settings for system
__________  RESIDENT CLISTS      Select resident clists for system
__________  TAPE ATTENDANCE      Select tape attendance for system
__________  APPLICATION DEFAULTS Select application defaults for system
__________  SYSTEM DEFAULTS      Select system defaults for system
__________  MVSCOMP DEFAULTS     Select MVS Component Defaults for system
__________  MVS COMPONENT        Select MVS Components for system
__________  NETWORK              Select network for system
__________  AUTOMATION OPERATORS Select automation operators for system
__________  Status Display       Select screen status definitions
__________  NOTIFY OPERATORS     Select notify operators
__________  IMS XRF RESOURCES    Select IMS XRF resource names for system
__________  IMS XRF STATUS FILES Select IMS XRF Status files for system
__________  OPC SYSTEM DETAILS   Select OPC system details for system
__________  CONTROLLER DETAILS   Select OPC controller details for system
__________  WORKSTATION DOMAINS  Select OPC Workstation domains for system
__________  -------------------- -----PROCESSOR OPERATIONS SPECIFIC POLICY----
__________  PROCESSOR            Select processor or partition
__________  TARGET SYSTEM INFO   Basics, time zones, and interested operators
__________  IPL INFO             Define IPL information
__________  -------------------- --------------------------------------------
__________  WHERE USED           List groups/subgroups linked to this entry
__________  COPY                 Copy data from existing entry
******************************* Bottom of data ********************************
```

Figure 87. Policy Selection Panel for Systems

The policy items for systems are:

SYSTEM INFO
This displays and allows you to update the following:

- The operating system type for this system definition.
- The heartbeat interval and the missing heartbeat delay times. Refer to "System Policy Object" on page 118 for more information.
- Details required for interacting with NetView. This information is required only if you are using enterprise monitoring.
- Details of your operating environment.

AUTOMATION SYMBOLS
You can enter or update SA z/OS symbols that can be set to various system-specific values. These can then be substituted into several values, such as MVS automatic restart management element names, MVS workload manager resource names, job names, user specified commands and others. Refer to "AUTOMATION SYMBOLS Policy Item" on page 127 for more information.

AUTOMATION CONSOLE
The automation console definitions are used when SA z/OS notification messages are to be sent as write-to-operator (WTO) messages to the MVS

consoles. This policy item defines the route codes and MCSFLAGS necessary for this process. WTO messages are sent to the Automation Console when none of the notification operators defined in the *Notify Operators* policy item are logged on or when SYSOP is defined as a notification operator.

For more information on this policy item refer to “AUTOMATION CONSOLE Policy Item” on page 128.

APPLICATION GROUPS
When you select this policy item, a list of all System Application Groups is displayed that indicates those that are linked to this system. You can add application groups to or remove them from the system.

MONITOR RESOURCES
This policy item allows you to manage links to Monitor Resource entries. Refer to “Monitor Resource Policy Object” on page 249 for more information.

AUTOMATION TIMERS
This policy item displays a list of defined Automation Timer sets and allows you to choose which sets are to be associated with this system object. See “Timers Policy Object” on page 274 for more information.

USER E-T PAIRS
This policy item displays a list of defined User E-T Pairs sets and allows you to choose which sets are to be associated with this system object. See “User E-T Pairs Policy Object” on page 313 for more information.

AUTOMATION TIMEOUT
This policy item displays a list of defined Automation Timeout sets and allows you to choose which sets are to be associated with this system object. See “Timeout Settings Policy Object” on page 277 for more information.

RESIDENT CLISTS
This policy item displays a list of defined Resident CLIST sets and allows you to choose which sets are to be associated with this system object. See “Resident CLISTs Policy Object” on page 309 for more information.

TAPE ATTENDANCE
This policy item displays a list of defined Tape Attendance sets and allows you to choose which sets are to be associated with this system object. See “Tape Attendance Policy Object” on page 279 for more information.

APPLICATION DEFAULTS
This policy item displays a list of defined Application Defaults sets and allows you to choose a set to be associated with this system object. If an application belongs to an application class, the class inheritance takes precedence. See “Application Defaults Policy Object” on page 290 for more information.

SYSTEM DEFAULTS
This policy item displays a list of defined System Defaults sets and allows you to choose a set to be associated with this system object. If an application belongs to an application class, the class inheritance takes precedence. See “System Defaults Policy Object” on page 288 for more information.

MVSCOMP DEFAULTS
This policy item displays a list of defined MVSCOMP Defaults sets and

allows you to choose a set to be associated with this system object. See "MVSCOMP Defaults Policy Object" on page 287 for more information.

MVS COMPONENT
This policy item displays a list of defined MVS Component sets and allows you to choose which sets are to be associated with this system object. See "MVS Component Policy Object" on page 282 for more information.

NETWORK
This policy item displays a list of defined Network sets and allows you to choose which sets are to be associated with this system object. See "Network Policy Object" on page 299 for more information.

AUTOMATION OPERATORS
This policy item displays a list of defined Automation Operator sets and allows you to choose which sets are to be associated with this system object. See "Automation Operators Policy Object" on page 291 for more information.

STATUS DISPLAY
This policy item displays a list Status Display Facility (SDF) Status Definition sets and allows you to choose which sets are to be associated with this system object. See "Status Display Policy Object" on page 310 for more information.

NOTIFY OPERATORS
This policy item displays a list of defined notification operator sets who receive SA z/OS notification messages.

See "Notify Operators Policy Object" on page 296 for more information.

IMS XRF RESOURCES
This policy item allows you to manage links to Extended Recovery Facility (XRF) resources. See "IMS Resource Name Policy Object" on page 263 for more information.

IMS XRF STATUS FILES
This policy item allows you to manage links to Extended Recovery Facility (XRF) status files. See "IMS Status Files Policy Object" on page 262 for more information.

Note that this is only for systems that are running releases from SA z/OS 2.3 or earlier.

OPC SYSTEM DETAILS
This policy item allows you to manage links to OPC System Details. Only one such entry should be linked to a system, otherwise multiple entries will be built with SA z/OS only loading the last one processed. See "OPC System Detail Policy Object" on page 264 for more information.

CONTROLLER DETAILS
This policy item allows you to manage links to OPC Controller Details. Only one such entry should be linked to a system, otherwise multiple entries will be built with SA z/OS only loading the last one processed. See "OPC Controller Details Policy Object" on page 266 for more information.

WORKSTATION DOMAINS
This policy item allows you to manage links to workstation domain IDs. Only one such entry should be linked to a system, otherwise multiple entries will be built with SA z/OS only loading the last one processed. See "OPC Workstation Domains Policy Object" on page 269 for more information.

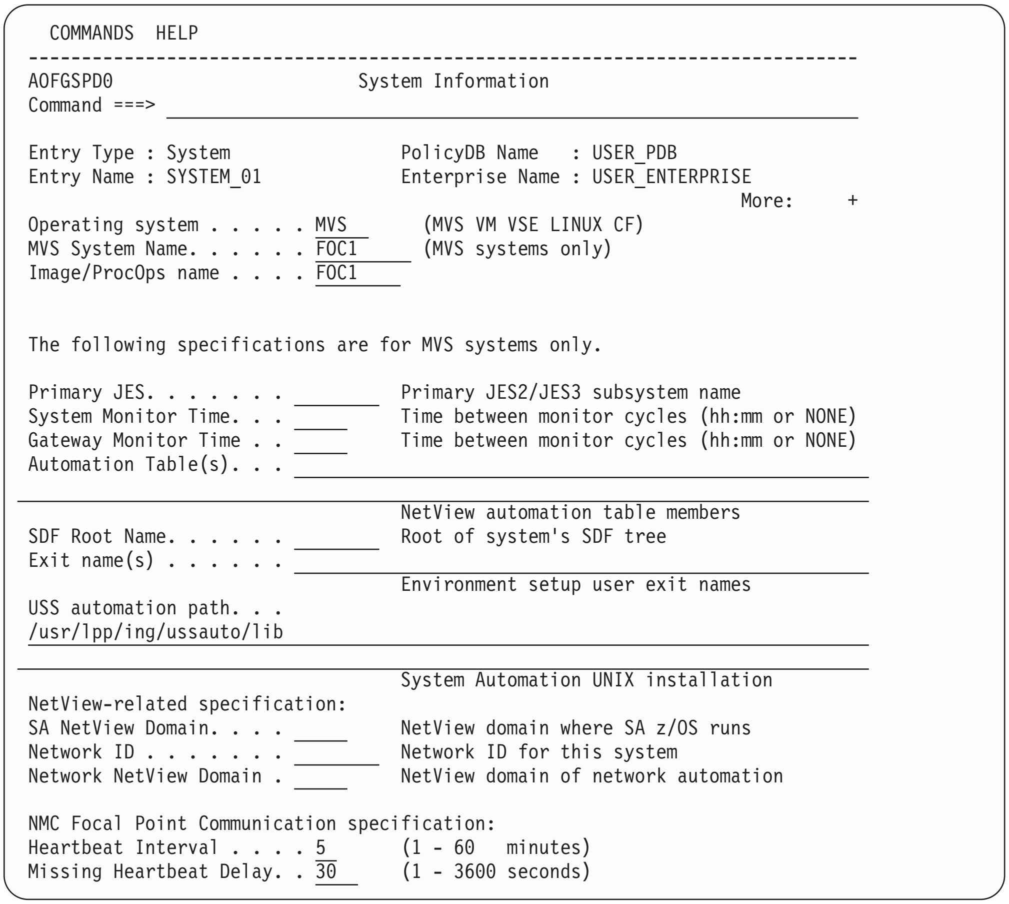
PROCESSOR
Selection of this policy item displays a panel that lists all processors and partitions defined in the enterprise. You can select one processor or partition that the system should run on.

TARGET SYSTEM INFO
This policy item displays basic information about the target system, for example, time zones, operators receiving messages from the system and operator consoles.

IPL INFO
With this policy item you can define IPL information for the MVS or VM target system.

SYSTEM INFO Policy Item

The panel shown in Figure 88 is displayed when you select policy item SYSTEM INFO on the Policy Selection panel for Systems (Figure 87 on page 122).



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGSPD0                       System Information
Command ===> _________________________________________________________________

Entry Type : System               PolicyDB Name   : USER_PDB
Entry Name : SYSTEM_01            Enterprise Name : USER_ENTERPRISE
                                                             More:     +
Operating system . . . . . MVS      (MVS VM VSE LINUX CF)
MVS System Name. . . . . . FOC1     (MVS systems only)
Image/ProcOps name . . . . FOC1


The following specifications are for MVS systems only.

Primary JES. . . . . . . ________   Primary JES2/JES3 subsystem name
System Monitor Time. . . _____      Time between monitor cycles (hh:mm or NONE)
Gateway Monitor Time . . _____      Time between monitor cycles (hh:mm or NONE)
Automation Table(s). . . ______________________________________________________
_______________________________________________________________________________
                                    NetView automation table members
SDF Root Name. . . . . . ________   Root of system's SDF tree
Exit name(s) . . . . . . ______________________________________________________
                                    Environment setup user exit names
USS automation path. . .
/usr/lpp/ing/ussauto/lib_______________________________________________________
_______________________________________________________________________________
                                    System Automation UNIX installation
NetView-related specification:
SA NetView Domain. . . . _____      NetView domain where SA z/OS runs
Network ID . . . . . . . ________   Network ID for this system
Network NetView Domain . _____      NetView domain of network automation

NMC Focal Point Communication specification:
Heartbeat Interval . . . . 5        (1 - 60   minutes)
Missing Heartbeat Delay. . 30       (1 - 3600 seconds)
```

Figure 88. System Information Panel

The following information can be entered:

Operating system
Identifies the operating system that runs on the system. Valid values are MVS, VM, VSE, LINUX and CF. CF indicates the coupling facility control code for the coupling facility in a Parallel Sysplex. LINUX indicates the Linux for zSeries operating system distributed for 390-CMOS and zSeries processors. This field is required and its default value is MVS.

MVS System Name

This is required for MVS systems. This value *must match* the system name (SYSID) as specified in the IEASYS*xx* member of SYS1.PARMLIB. Valid characters are any alphabetic or numeric character or any of @, # or $. Note that the first three characters of the MVS SYSNAME must not be ″AOF″, as these are reserved for internal SA z/OS processing. To define a system for an alternate configuration, it must be marked with a trailing ″+″ in its MVS SYSNAME. This trailing character will be removed when the automation files are built.

If you are going to enable automation for IXC102A/IXC402D the MVS SYSNAME must not start with a numeric character. For more information see *IBM Tivoli System Automation for z/OS Customizing and Programming*.

Image/ProcOps name

Specifies the name of the system for hardware related functions, for example, processor operations functions. As the entry name for the system, this name must be unique among the names of all groups, subgroups, systems, processors, and image names defined in the current policy database. For MVS systems, the name is defaulted to the MVS SYSNAME but it can be changed. Valid characters are any alphabetic or numeric character, or any of @, # or $.

Primary JES

The name of the primary Job Entry Subsystem as specified for a JES2 or JES3 subsystem. The named subsystem is the default scheduling subsystem.

System Monitor Time

The time between system monitoring cycles. The default is every 59 minutes. Monitoring may be suspended by specifying NONE.

Gateway Monitor Time

The time between gateway monitoring cycles. The default is 15 minutes. Monitoring may be suspended by specifying NONE.

Automation Table

You may specify multiple automation tables that are loaded and used for message automation. The automation tables are activated in the sequence as specified. The sequence of the automation tables is not changed during a configuration refresh (INGAMS REFRESH).

The first automation table of the list defined here replaces the SA z/OS initial automation table INGMSG00 that was loaded during initialization. The default is INGMSG01.

SDF Root Name

Defines the name used by SDF for the system name. This is the root component that is used during initialization. If nothing is entered the default is the **MVS System Name**

Exit name(s)

An installation exit or exits invoked to perform additional environment setup. This item is optional. Refer to *IBM Tivoli System Automation for z/OS Customizing and Programming* for more information.

USS automation path

Defines the path where the SA z/OS USS automation programs (for example, INGCCMD) are installed. The SA z/OS installation default is /usr/lpp/ing/ussauto/lib.

SA NetView Domain
This field is optional. It allows you to enter the NetView domain name of the NetView that the SA z/OS functions run under on this system. This information is applicable only if you are using a dual NetView setup. The SA z/OS NetView name can be up to five characters long.

Network ID
This ID is required only if you are using enterprise monitoring. It specifies the network that the system operations NetView domain is located in. It can be up to eight characters long.

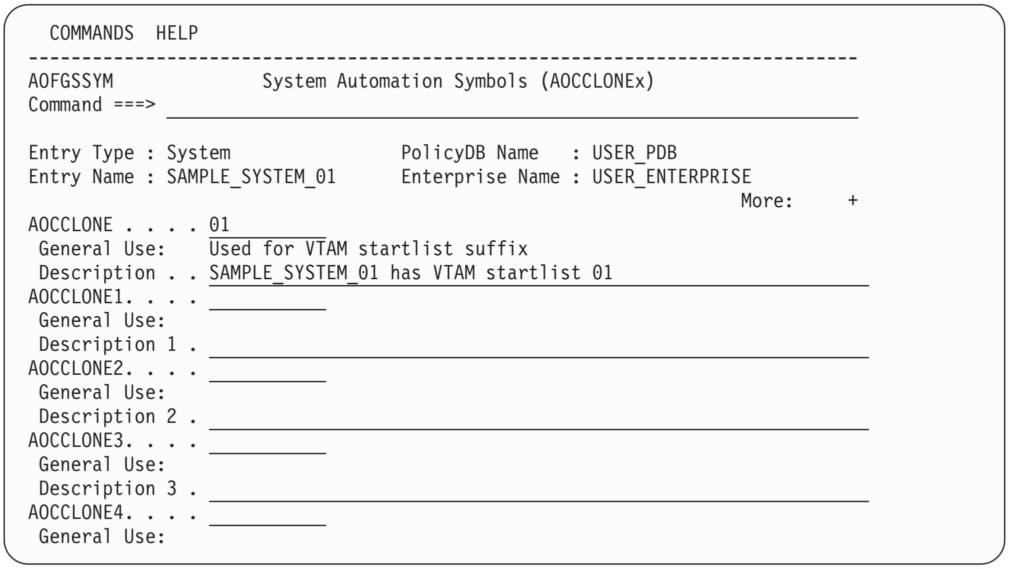
Network NetView Domain
This field is optional. It identifies the NetView domain that runs network automation. It can be up to five characters long. When the automation control file is loaded, this value is assigned to the common global variable AOF_NETWORK_DOMAIN_ID.

Heartbeat Interval
Specifies the time interval at which the SA z/OS system operations functions will send a generic alert from this system to the NMC Focal Point. The SA z/OS system operations functions on the NMC Focal Point use heartbeat alerts to verify the status forwarding path from each remote system. The heartbeat interval is a required field in the range of 1 to 60 minutes. The default is 5 minutes. Heartbeat information is only needed for MVS systems.

Missing Heartbeat Delay
Specifies the time that the SA z/OS system operations functions on the NMC Focal Point will wait after a heartbeat was expected. If this time expires without receiving the heartbeat, system operations will begin missing heartbeat processing for the remote system. The field is provided as a way to buffer against temporary network delays. The missing heartbeat delay is a required field in the range of 1 to 3600 seconds. 30 seconds is the default. Heartbeat information is only needed for MVS systems.

AUTOMATION SYMBOLS Policy Item

The panel shown in Figure 89 on page 128 is displayed when you select policy item AUTOMATION SYMBOLS from the Policy Selection panel for Systems (Figure 87 on page 122).



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGSSYM              System Automation Symbols (AOCCLONEx)
Command ===>

Entry Type : System               PolicyDB Name   : USER_PDB
Entry Name : SAMPLE_SYSTEM_01     Enterprise Name : USER_ENTERPRISE
                                                                More:     +
AOCCLONE . . . . 01
 General Use:    Used for VTAM startlist suffix
 Description . . SAMPLE_SYSTEM_01 has VTAM startlist 01
AOCCLONE1. . . .
 General Use:
 Description 1 .
AOCCLONE2. . . .
 General Use:
 Description 2 .
AOCCLONE3. . . .
 General Use:
 Description 3 .
AOCCLONE4. . . .
 General Use:
```

Figure 89. Automation Symbols Panel

This panel allows you to enter or update system automation symbols that can be set to various system-specific values. These can then be substituted into several values, such as MVS automatic restart management element names, MVS workload manager resource names, job names, user-specified commands and others.

To specify that AOCCLONE value `x` is to be substituted, enter `&AOCCLONEx.` at the appropriate point in the field. You must specify both the leading ampersand (&) and the trailing period (.).

You may, additionally, specify the tilde (~) to get the 0th (unnumbered) AOCCLONE value substituted. For more details refer to “Using System Symbols and System Automation Symbols” on page 28.

This panel has the following fields:

General Use
This shows the description of the general use of the system automation symbol. It is defined in the SYMBOL DESCRIPTION policy item of the enterprise definitions (entry type ENT), see “SYMBOL DESCRIPTION Policy Item” on page 97.

AOCCLONE*x*
This allows you to specify the value of the system automation symbol (&AOCCLONE*x* variable).

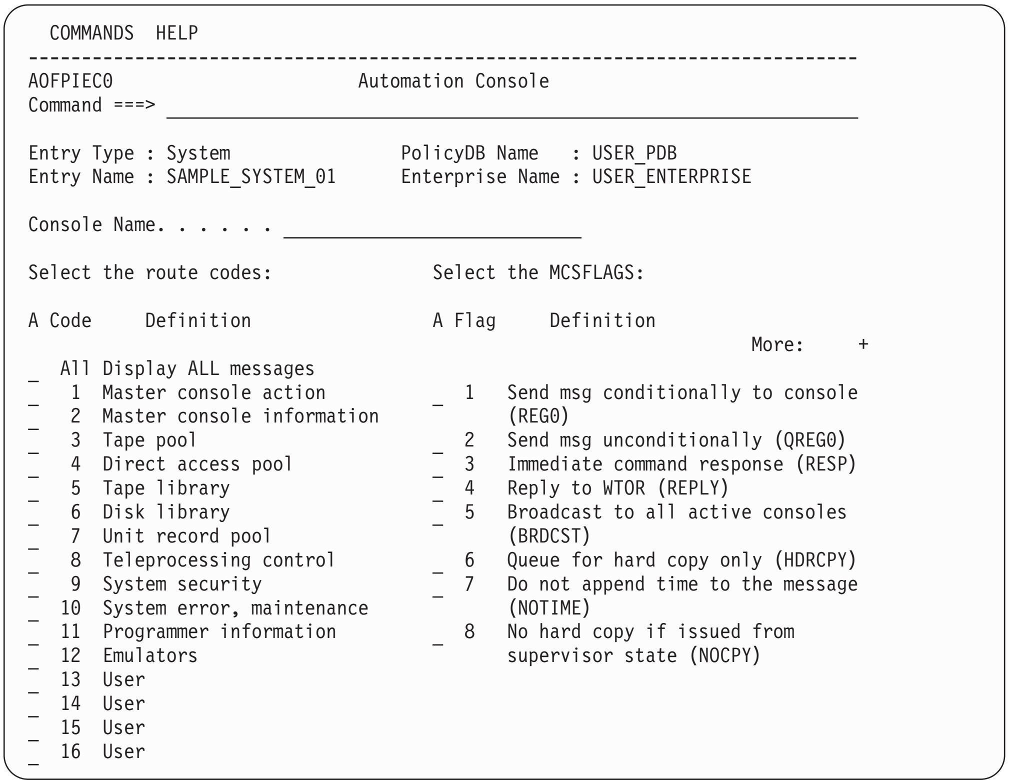
Note: Valid characters are A–Z, a–z, 0–9, @ (X'7C'), # (X'7B'), $ (X'5B'), ? (X'6F'), _ (X'6D'), ! (X'5A'), and ¢ (X'4A'). The entered data is not automatically translated to upper case.

Description
This allows you to specify a specific description of the automation symbol for this system only.

AUTOMATION CONSOLE Policy Item

The panel shown in Figure 90 on page 129 is displayed when you select policy item AUTOMATION CONSOLE from the Policy Selection panel for Systems

(Figure 87 on page 122).



```
   COMMANDS  HELP
 ------------------------------------------------------------------------------
 AOFPIEC0                       Automation Console
 Command ===> _________________________________________________________________

 Entry Type : System               PolicyDB Name   : USER_PDB
 Entry Name : SAMPLE_SYSTEM_01     Enterprise Name : USER_ENTERPRISE

 Console Name. . . . . . ____________________________

 Select the route codes:              Select the MCSFLAGS:

 A Code     Definition                A Flag     Definition
                                                                    More:     +
 _  All Display ALL messages
 _   1  Master console action         _  1   Send msg conditionally to console
 _   2  Master console information            (REG0)
 _   3  Tape pool                     _  2   Send msg unconditionally (QREG0)
 _   4  Direct access pool            _  3   Immediate command response (RESP)
 _   5  Tape library                  _  4   Reply to WTOR (REPLY)
 _   6  Disk library                  _  5   Broadcast to all active consoles
 _   7  Unit record pool                     (BRDCST)
 _   8  Teleprocessing control        _  6   Queue for hard copy only (HDRCPY)
 _   9  System security               _  7   Do not append time to the message
 _  10  System error, maintenance            (NOTIME)
 _  11  Programmer information        _  8   No hard copy if issued from
 _  12  Emulators                            supervisor state (NOCPY)
 _  13  User
 _  14  User
 _  15  User
 _  16  User
```

Figure 90. Environment Definition: CONSOLE (Route Codes) Panel

In order to define an automation console, you need to specify:

Console Name
This parameter defines the 1-8 alphanumeric console name specified in the CONSOL*nn* member of the z/OS PARMLIB.

Route Codes
The appropriate z/OS route codes can be selected from this panel. These route codes define which kind of messages the automation console is to receive. Each notification message has already been defined as one of these kinds of WTOs. The console can also be defined to receive all of the notification messages regardless of how they have been defined. The last four codes on the panel are reserved for user-defined codes.

The default routing code is 1 (only messages that appear as Master Console Action WTOs are displayed on the console). Further information on the route codes can be found in the MVS System Messages books.

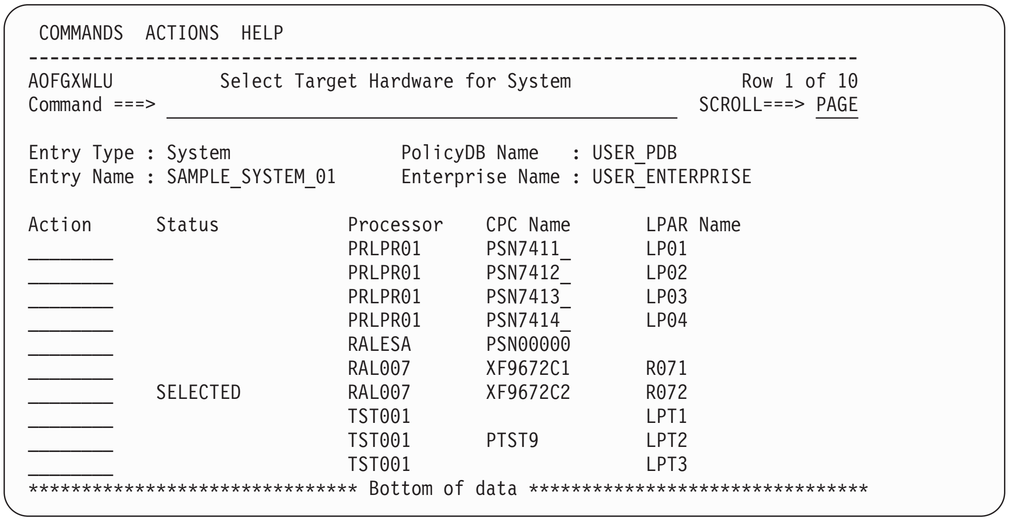
MCSFLAGS
Select the appropriate MVS MCSFLAG from the list. These codes are used to provide additional information for the WTO messages (for example, if flag 7 is set, the time is not included with the WTO).

If flag 1 or 2 is set, the console name is required and the messages are queued to the specified console.

Further information on the MCSFLAG codes can be found in *Tivoli NetView for z/OS Customization: Using REXX and the NetView Command List Language*

PROCESSOR Policy Item

The panel shown in Figure 91 is displayed when you select policy item PROCESSOR on the Policy Selection panel for Systems (Figure 87 on page 122).



```
 COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGXWLU         Select Target Hardware for System              Row 1 of 10
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : System               PolicyDB Name   : USER_PDB
Entry Name : SAMPLE_SYSTEM_01     Enterprise Name : USER_ENTERPRISE

Action      Status           Processor    CPC Name       LPAR Name
________                     PRLPR01      PSN7411_       LP01
________                     PRLPR01      PSN7412_       LP02
________                     PRLPR01      PSN7413_       LP03
________                     PRLPR01      PSN7414_       LP04
________                     RALESA       PSN00000
________                     RAL007       XF9672C1       R071
________    SELECTED         RAL007       XF9672C2       R072
________                     TST001                      LPT1
________                     TST001       PTST9          LPT2
________                     TST001                      LPT3
******************************* Bottom of data ********************************
```

Figure 91. Select Target Hardware for System Panel

This panel allows you to link a system to the target hardware; it displays a list of:

- All processors together with their LPARs.
- All processors in basic mode.

The panel has the following unique fields:

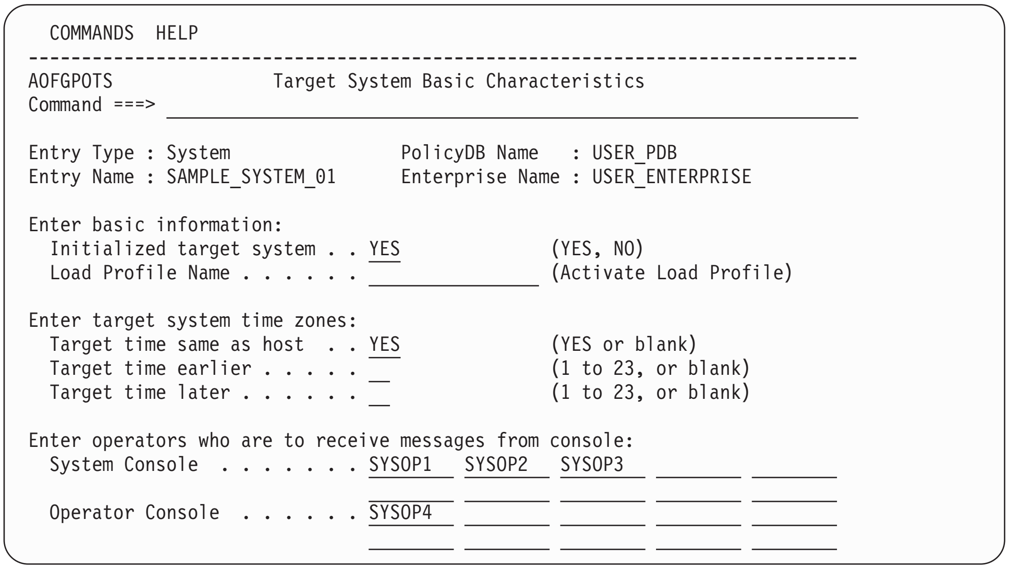
Processor
Shows the processor entry name.

CPC Name
Shows the CPC Name/NAU as specified on the Processor Information panel. This field may be blank if data for the Processor Information policy has not yet been specified.

LPAR Name
Shows the name of the logical partition as specified on the LPAR Definitions panel. This field is empty for basic processors .

TARGET SYSTEM INFO Policy Item

The panel shown in Figure 92 on page 131 is displayed when you select policy item TARGET SYSTEM INFO on the Policy Selection panel for Systems (Figure 87 on page 122).



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGPOTS              Target System Basic Characteristics
Command ===> _________________________________________________________________

Entry Type : System                PolicyDB Name   : USER_PDB
Entry Name : SAMPLE_SYSTEM_01      Enterprise Name : USER_ENTERPRISE

Enter basic information:
  Initialized target system . . YES              (YES, NO)
  Load Profile Name . . . . . . ________________ (Activate Load Profile)

Enter target system time zones:
  Target time same as host  . . YES              (YES or blank)
  Target time earlier . . . . . __               (1 to 23, or blank)
  Target time later . . . . . . __               (1 to 23, or blank)

Enter operators who are to receive messages from console:
  System Console  . . . . . . . SYSOP1__ SYSOP2__ SYSOP3__ ________ ________
                                ________ ________ ________ ________ ________
  Operator Console  . . . . . . SYSOP4__ ________ ________ ________ ________
                                ________ ________ ________ ________ ________
```

Figure 92. Target System Information Panel

Initialized Target System
Enter YES to indicate that the target system is automatically initialized when SA z/OS processor operations starts. Otherwise, enter NO.

> **Note:**
> If you choose to initialize the target system automatically, be aware that when you start SA z/OS, the ISQSTART command may be completed before the target system is initialized.

Load Profile Name
Enter the load profile name defined at the support element. The name you choose must correspond with the load profile defined at the support element. Leave this field blank to take the default defined at the support element.

> **Note:**
> This profile name can contain only the characters A to Z and 0 to 9.

Target time same as host
If the target system is in the same time zone as the focal-point system, type YES or Y. If not, leave this field blank. For systems that run as guest systems under VM this field must be set to YES.

Target time earlier
Enter the number of hours that the target system is earlier than the focal point system. (A target system in Los Angeles is three hours earlier than a focal-point system in New York.)

Target time later
Enter the number of hours that the target system is later than the focal point system. (A target system in New York is three hours later than a focal point system in Los Angeles.)

System Console
Enter the user IDs of the operators you want to receive system console

messages. The operators listed receive messages only from the system console. You can enter any valid NetView user ID. This field is optional.

Operator Console
Enter the user IDs of the operators you want to receive operator console messages. You can enter any valid NetView user ID. This field is optional.

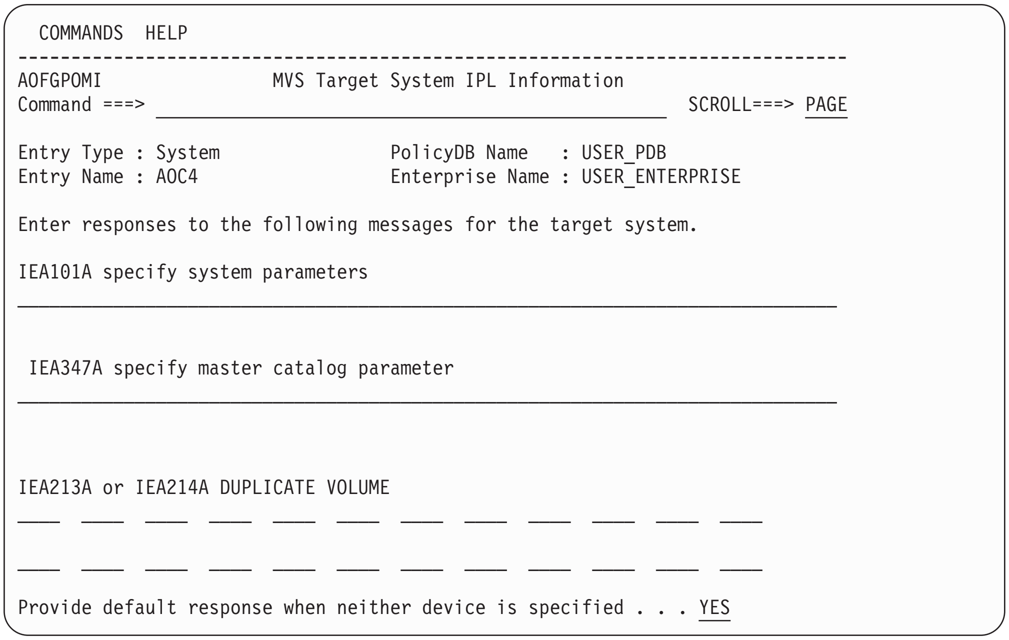
> **Notes:**
>
> 1. For OCF-based systems using only console integration, the operating system messages are forwarded to the system console interested operator list.
> 2. Operators not on these lists do not see messages from these consoles unless their user IDs are added using the ISQXMON command when SA z/OS is running.

IPL INFO Policy Item

The panels you can use to define your target system IPL information depends on the operating system you are running.

MVS Target System IPL Information

The panel shown in Figure 93 is displayed when you select policy item IPL INFO on the Policy Selection panel for Systems (Figure 87 on page 122) for MVS.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGPOMI                MVS Target System IPL Information
Command ===> ______________________________________________  SCROLL===> PAGE

Entry Type : System                PolicyDB Name   : USER_PDB
Entry Name : AOC4                  Enterprise Name : USER_ENTERPRISE

Enter responses to the following messages for the target system.

IEA101A specify system parameters
_____________________________________________________________________________


 IEA347A specify master catalog parameter
_____________________________________________________________________________



IEA213A or IEA214A DUPLICATE VOLUME
____  ____  ____  ____  ____  ____  ____  ____  ____  ____  ____  ____

____  ____  ____  ____  ____  ____  ____  ____  ____  ____  ____  ____

Provide default response when neither device is specified . . . YES
```

Figure 93. MVS Target System IPL Information Panel

Use this panel to define responses used during the MVS or z/OS target IPL process.

During the start process, the target system requests information from the operator. By providing the responses at this panel, the operator enables SA z/OS to respond to these messages. Target systems using VSE, LINUX, or CF do not request information from the operator during startup.

IEA101A specify system parameters

Enter the desired response to the IEA101A message. All IEA101A message responses begin with the characters R 0. You do not have to begin your response with these characters, they are provided by SA z/OS automation. Quotation marks are not necessary. The customization dialog does not verify your entry.

IEA347A SPECIFY MASTER CATALOG PARAMETER

Enter the desired response to the IEA347A message. The customization dialog does not verify your entry.

IEA212A or IEA213A/IEA214A duplicate volume

Specify the device addresses to be kept online if duplicate DASD volumes exist on a z/OS system at IPL time. You have 24 entries to specify real or generic DASD device addresses. Example for a generic device address: `012*`. This indicates that all addresses from 0120 to 012F must remain online.

This entry works in conjunction with the automatic response selection.

Provide default response when neither device is specified. Specify YES or NO to indicate if a default response is required if neither or both of the devices in the IEA213A/IEA214A message match those identified as required devices. For IEA214A messages, the current IPL (SYSRES) device is considered as being implicitly defined. If YES is selected, an automated response will always be provided. For a SYSRES device, the device shown in the message is dismounted. Otherwise the following applies:

- If neither device is in the reply list, the first device is demounted.
- If only one device is in the reply list, the remaining device is demounted.
- If both devices are in the reply list, the first device in the list as scanned from left to right remains mounted and the other device is demounted.
- If both devices are in the reply list due to the same generic device mask in the list as scanned from left to right, the second device is demounted.

A selection of NO will result in a response allowing the IPL to continue only if one and only one device is identified as being required. Otherwise, message ISQ1203 will be issued and the IPL process will be suspended. In such a case, the operator needs to react to system message ISQ1203.

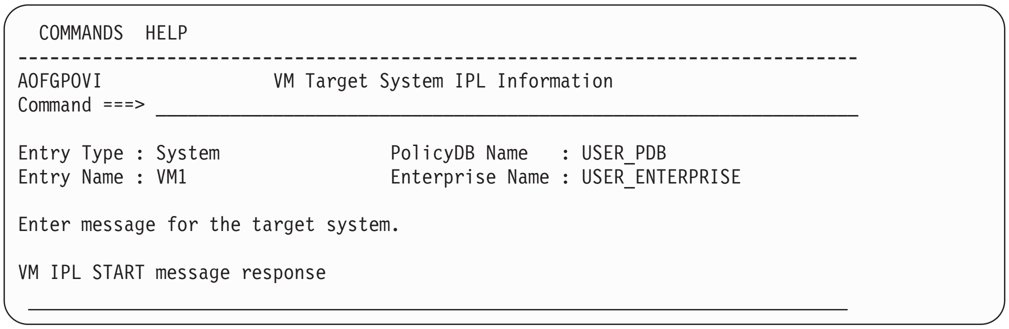
A selection of NO and a generic device address of **** will always stop the IPL process if a duplicate volume is detected.

Notes:

1. For valid responses, see *z/OS MVS System Messages, Vol 3 (ASB-BPX)*.
2. Responses to IEA101A, IEA347A, IEA212A, and IEA213A can be changed without generating and loading a new SA z/OS processor control file by the use of the host ISQXOPT command before IPLing the target system. However, these changes are not saved if SA z/OS processor is stopped or restarted.
3. You receive either message IEA212A or IEA213A depending on the level of your MVS system.
4. All message responses begin with the characters R 0. You do not have to begin your response with these characters; SA z/OS automation provides them. Quotation marks are also not necessary.

VM Target System IPL Information

To define the response to the VM IPL START message, use the panel shown in Figure 94. During the start process, the target system requests information from the operator. By providing the responses at this panel, the operator enables SA z/OS to respond to these messages.



```
  COMMANDS  HELP
-------------------------------------------------------------------------------
AOFGPOVI                VM Target System IPL Information
Command ===> __________________________________________________________________

Entry Type : System                PolicyDB Name   : USER_PDB
Entry Name : VM1                   Enterprise Name : USER_ENTERPRISE

Enter message for the target system.

VM IPL START message response
 ______________________________________________________________________________
```

Figure 94. VM Target System IPL Information Panel

VM IPL START message response
Enter the desired response to the VM IPL START message. The customization dialog does not verify your entry.

Note:
The VM IPL START message response field can be changed without generating and loading a new processor operations control file. You can do this by issuing the host ISQXOPT command before you IPL the target system. However, these changes are not saved if processor operations or SA z/OS is stopped or restarted.

ApplicationGroup Policy Object

An application group is a set of applications that you assign a name to. There are two types of ApplicationGroups:

1. *System Application Groups* - application groups that are associated with particular systems.
2. *Sysplex Application Groups* - application groups that are associated with sysplexes.

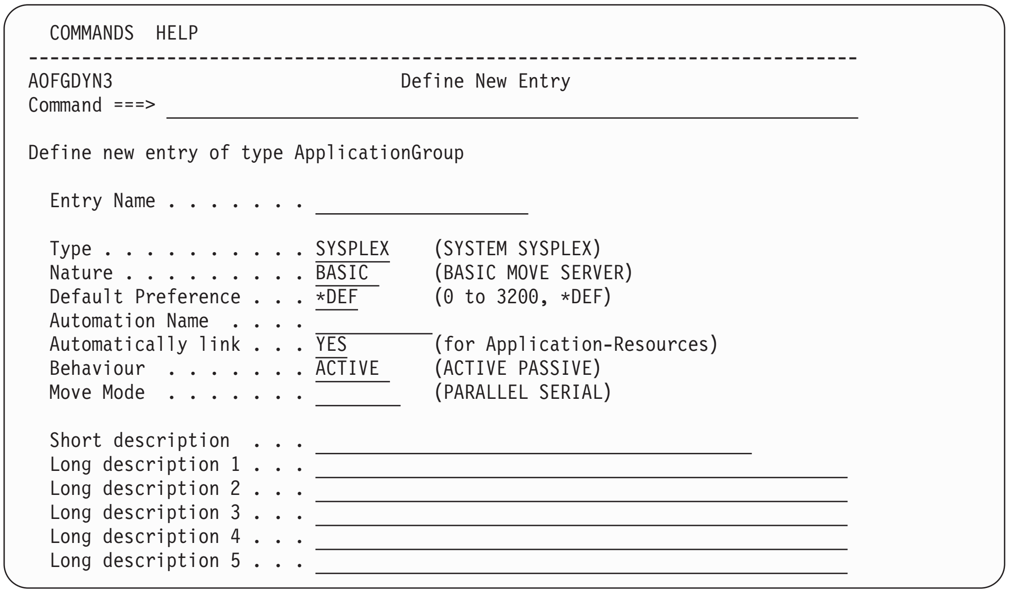
A *Sysplex Application Group* is a special type of application group that is associated with one or more sysplexes. Sysplex Application Groups are used to monitor and automate applications and application groups that can run on multiple systems in a sysplex. Using Sysplex Application Groups means you do not need to connect the application group to each system of the sysplex. Instead, all applications of this Sysplex Application Group are known on all systems of the sysplex. You can then monitor the group at the sysplex level, rather than at system level.

The SA z/OS customization dialog allows concurrent user access capability for entry type ApplicationGroup. See also Appendix A, “Concurrent Customization by Multiple Users,” on page 387 for further information.

Creating a New ApplicationGroup

How to create new policy objects in general, is described in “How to Create New Policy Objects” on page 37.

Figure 95 shows the panel that is displayed when you create a new application group with the *New* command from the Entry Name Selection panel for application groups.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN3                           Define New Entry
Command ===>

Define new entry of type ApplicationGroup

  Entry Name . . . . . . .

  Type . . . . . . . . . . SYSPLEX    (SYSTEM SYSPLEX)
  Nature . . . . . . . . . BASIC      (BASIC MOVE SERVER)
  Default Preference . . . *DEF       (0 to 3200, *DEF)
  Automation Name  . . . .
  Automatically link . . . YES        (for Application-Resources)
  Behaviour  . . . . . . . ACTIVE     (ACTIVE PASSIVE)
  Move Mode  . . . . . . .            (PARALLEL SERIAL)

  Short description  . . .
  Long description 1 . . .
  Long description 2 . . .
  Long description 3 . . .
  Long description 4 . . .
  Long description 5 . . .
```

Figure 95. Define New Entry Panel for a Sysplex Application Group

The unique fields are:

Name This is the name of the application group. It must be unique within the policy database. Valid names may contain up to twenty alphanumeric characters.

Application Group Type

This indicates whether the Application Group will be used for Systems or for a Sysplex. This attribute sets the boundaries for the selection of Resource members of the Application Group, and should be set as follows:

SYSPLEX

Use this type if the Application Group is being used to define sets of Applications and Resources that may be on multiple systems within a sysplex or sysplexes.

SYSTEM

Use this type if the Application Group is being used to define sets of Applications and Resources that be on single systems.

Application Group Types cannot be changed after this panel has been left.

Nature

In this field you can define the availability behavior of application groups and applications. You can specify one of the following attributes:

BASIC

The group is available when all of its resource members are available.

MOVE

The group is available when one of its resource members is available.

All members should be instances of the same application.

To avoid the following conflict, you should not define an application to be a member in two MOVE groups: Assume that resource RES1 is a member in two MOVE groups. One of the MOVE groups starts a member which is not shared by the other MOVE group. This implies that RES1 should be down for this MOVE group, because only one member should be available. The second MOVE group now starts the shared member RES1. This causes the first MOVE group to run with two applications. The move group will ensure that only one member of the group will be active at any point in time.

SERVER

This is like a move group. Additionally, you need to define an *availability target* that determines how many of its members should be available to define the server group as being available, as well as a *satisfactory target* that specifies the number of members in the group that must be active before the automation manager treats the group as being in a satisfactory state.

All members should be instances of the same application.

Depending on your selection for the **Nature** field, your process of entering resources into a group will be different, as described in "How to Work with Resources" on page 47. Refer to this chapter for examples on how to use the different natures appropriately.

Default Preference

This field specifies what preference should be used for application resources when automatically linking applications to ApplicationGroups (note that the **Automatically link** field below must be set to YES for automatic linking and that preference values are only used for SERVER and MOVE ApplicationGroups). The *DEF preference corresponds to default preference 700. This is the default for this field. For information about preference values and their meanings refer to "RESOURCES Policy Item" on page 148 and to "Using Preferences to Define Specific Roles" on page 67. Or, for more details about how an operator can override preference values, refer to *IBM Tivoli System Automation for z/OS User's Guide*.

Automation Name

If you want to automate and monitor the current application group, you need to specify an automation name in this field.

At the time an application group is created, it is not yet known which systems you want to link it to. So you need to use its *WHERE USED* policy item to connect it to the systems you want. Suppose you created a System Application Group called SYST_APG and linked it to systems SYSTEM1 (with the same MVS SYSNAME) and SYSTEM2 (with the same MVS SYSNAME). After pressing the END key, SA z/OS generates the following resources:

```
SYST_APG/APG/SYSTEM1
SYST_APG/APG/SYSTEM2
```

If you do not provide an automation name, an ACF fragment is not built for the APG and thus the APG is not included in the ACF main include member (ACFZ*xxx*). If you change the automation name of an APG to or from blank (that is, no name is specified), you need to rebuild the main include member to reflect this change.

Automatically link

This entry field specifies whether applications that will later be linked to the current ApplicationGroup should also be automatically linked as resources. This field should be changed to NO if no more resources should automatically be added if the group is linked to another system or sysplex. Otherwise the automation behavior might not be as expected for complex group definitions (that is, groups with manually adapted preference values or availability targets other than *ALL).

Note: This field is only relevant for application groups that have a specified automation name. For others the default value YES is assumed.

Behaviour

With this field you can define whether requests to the application group should be propagated to the group members or not. With this option you can avoid having a resource unintentionally stopped if it is required for some other process outside the group.

ACTIVE

Propagates requests to all members of the application group according to the application group's nature, for example, a start request to a BASIC group is propagated to all members, as a BASIC group is considered to be available when all members are available. A start request to a MOVE group is forwarded to the member with the highest preference, as a MOVE group is considered to be available when exactly one member is available.

ACTIVE is the default.

PASSIVE

Does not propagate the requests to the group members. This enables monitoring and disables the automation of the application group.

Move Mode

Specifies how a move is made from an old member to a new one:

PARALLEL

The new member is started at the same time as the old member is being stopped. This is the default.

SERIAL

The new member is started after the old member has been stopped completely.

blank The value can be inherited from the system's default definition.

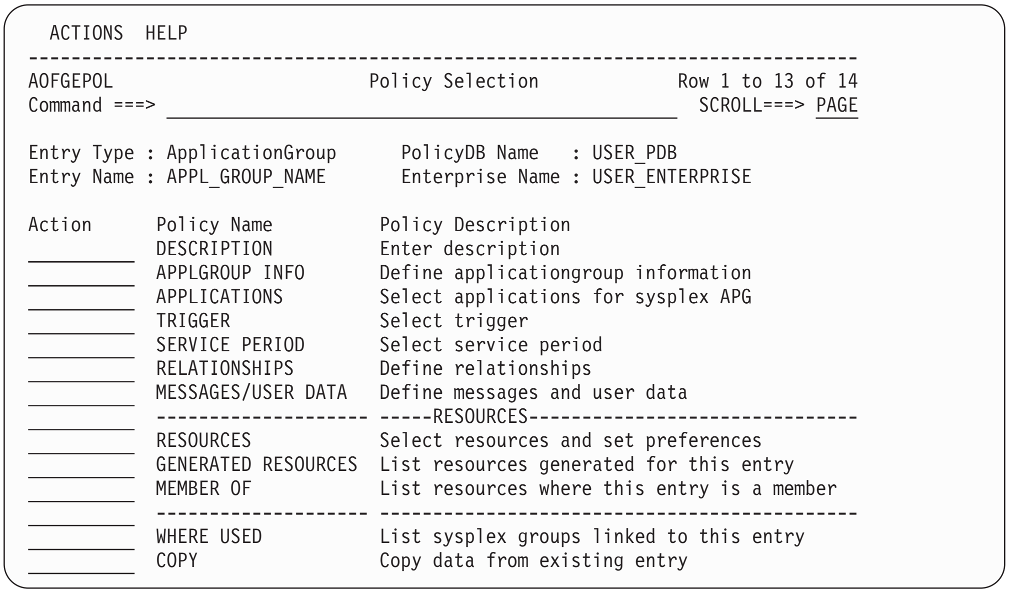
Note: This field is only relevant for groups of type MOVE. For groups of any other type it is forced to blank.

Policy Items for ApplicationGroups

As described in "Creating a New ApplicationGroup" on page 134, there are two types of application groups:

- System Application Groups
- Sysplex Application Groups

Figure 96 shows the policy items that are displayed when a System Application Group is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                      Policy Selection           Row 1 to 13 of 14
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : APPL_GROUP_NAME       Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  APPLGROUP INFO       Define applicationgroup information
__________  APPLICATIONS         Select applications for sysplex APG
__________  TRIGGER              Select trigger
__________  SERVICE PERIOD       Select service period
__________  RELATIONSHIPS        Define relationships
__________  MESSAGES/USER DATA   Define messages and user data
__________  -------------------- -----RESOURCES-------------------------------
__________  RESOURCES            Select resources and set preferences
__________  GENERATED RESOURCES  List resources generated for this entry
__________  MEMBER OF            List resources where this entry is a member
__________  -------------------- --------------------------------------------
__________  WHERE USED           List sysplex groups linked to this entry
__________  COPY                 Copy data from existing entry
```

Figure 96. Policy Selection Panel for System Application Groups

The unique policy items for application groups are:

APPLGROUP INFO
This policy item displays information about the Application Group, as specified when creating the application group. The panel is very similar to the one used for creating the application group. You may update indicated values in this panel. See “APPLGROUP INFO Policy Item” on page 139 for more information.

APPLICATIONS
When you select this policy item, a list of all applications in the SA z/OS enterprise is displayed, indicating which, if any, are associated with this application group. You can add applications to the application group or remove applications from the application group. If the current application group has a resource name, the appropriate APL resources are generated. According to the application group’s settings, the application resources (APL resources) are linked as members into the application group resource (APG resource). See “APPLICATIONS Policy Item” on page 141 for more information on this policy item.

TRIGGER
Each application group may be linked to one trigger. With this policy item, you can use the Trigger for ApplicationGroup panel to link the application group to a trigger so that all of its applications will be started or stopped depending on the defined trigger conditions.

SERVICE PERIOD
With this policy item, you can use the Service Period for ApplicationGroup panel to link a service period to the application group. Its resources are then automatically started up or shut down in the time frame specified by the service period.

RELATIONSHIPS
This policy item has the same functionality for entry types Application, ApplicationGroup, and *Monitor Resource*. For each application, application

group, or monitor resource you may specify several relationships. They will be used to automate the startup or stopping of those applications, application groups, and monitor resources. If you select this policy item, in the Relationship Selection List panel, you can view existing relationships or define new relationships for the current application, application group, or monitor resource.

You define relationships between dependent resources and supporting resources. In the customization dialog you define relationships always from the point of view of the dependent resource. So the resource that you have currently selected will become the dependent resource, when you define a new relationship.

Refer to “RELATIONSHIPS Policy Item” on page 143 for more information on viewing existing or defining new dependency relationships.

MESSAGES/USER DATA

This policy item has the same functionality as for applications and MVS components. It allows you to specify automated actions, such as the issuing of a command or reply, to be taken in response to a message. However, because application groups do not produce ’real’ messages, this policy behaves differently. Currently only the special message ID, INGALERT, together with the command definitions assigned to it are honored by SA z/OS.

How to proceed in specifying this policy item is described in “MESSAGES/USER DATA Policy Item” on page 283 where the policy is described for MVS components.

RESOURCES

This policy item allows you to select resources for current ApplicationGroup and set their preferences or name.

The panels displayed look different depending on the nature of the current application group. Refer to Chapter 4, “Scenarios on How to Use the Customization Dialog,” on page 47 and “RESOURCES Policy Item” on page 148 for information on how to work with these different panels.

GENERATED RESOURCES

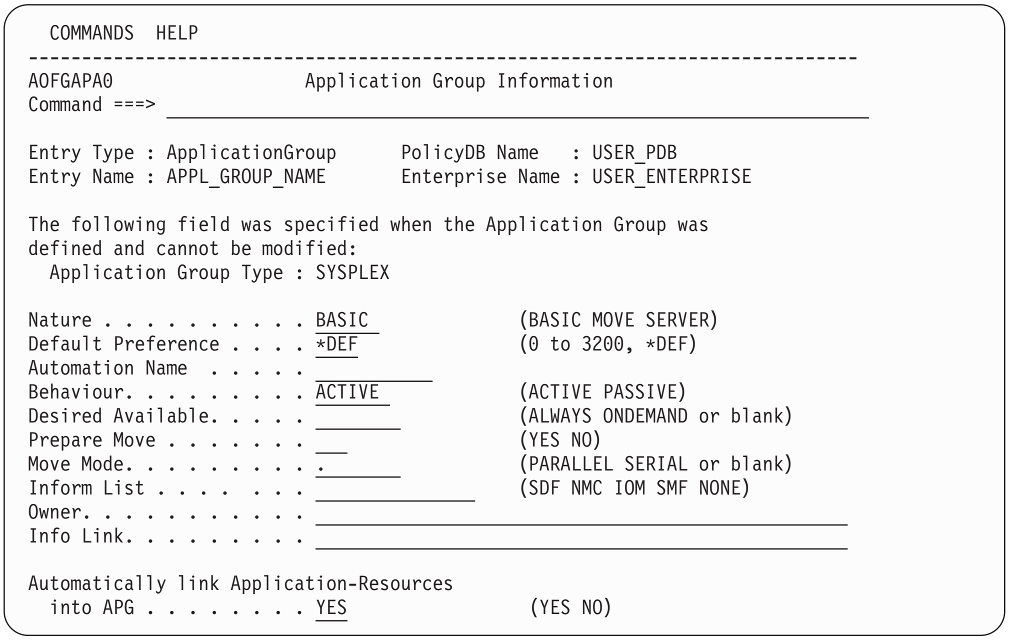
This policy item displays a list of all resources that have been generated for the current application group and for the applications that are linked to this application group. See “GENERATED RESOURCES Policy Item” on page 155 for a description of the Generated Resources panel.

MEMBER OF

As resources derived from application groups can be members of other APG resources, this policy item lets you see those resources that the current application group resource is a member of. Refer to panel Member of in “MEMBER OF Policy Item” on page 156 for more details.

APPLGROUP INFO Policy Item

A panel like Figure 97 on page 140 will be displayed if you select policy item APPLGROUP INFO from the Policy Selection Panel for Application Groups.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGAPA0                 Application Group Information
Command ===> ___________________________________________________________________

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : APPL_GROUP_NAME       Enterprise Name : USER_ENTERPRISE

The following field was specified when the Application Group was
defined and cannot be modified:
  Application Group Type : SYSPLEX

Nature . . . . . . . . . . . BASIC            (BASIC MOVE SERVER)
Default Preference . . . . *DEF             (0 to 3200, *DEF)
Automation Name  . . . . . ___________
Behaviour. . . . . . . . . ACTIVE           (ACTIVE PASSIVE)
Desired Available. . . . . ________         (ALWAYS ONDEMAND or blank)
Prepare Move . . . . . . . ___              (YES NO)
Move Mode. . . . . . . . . ._______         (PARALLEL SERIAL or blank)
Inform List . . . .  . . . _______________  (SDF NMC IOM SMF NONE)
Owner. . . . . . . . . . . __________________________________________________
Info Link. . . . . . . . . __________________________________________________

Automatically link Application-Resources
  into APG . . . . . . . . YES               (YES NO)
```

Figure 97. Application Group Information Panel

You can update the values shown in the lower part of this panel.

Nature
Refer to “Nature” on page 135 for more details.

Default Preference
Refer to “Default Preference” on page 136 for more details.

Automation Name
Refer to“Automation Name” on page 136 for more details.

Behaviour
Refer to “Behaviour” on page 137 for more details.

Desired Available
Specifies how the desired state of the group is determined in the absence of any propagated DesiredStatus votes on the resource.

You can specify the following values:

ALWAYS
The group's desired status is set to Available and it is started once its dependencies are satisfied. This is the same behavior that older releases of SA z/OS exhibit.

ONDEMAND
The group's desired status is set to Unavailable and it is not be started. For SA z/OS to start the group it must be propagated a MakeAvailable vote. This could come from an operator request, a service period, or a Move or Server group.

Note: A dependency from a Desired Available: ALWAYS resource is not sufficient to cause a Desired Available: ONDEMAND group to be started.

blank The value can be inherited from the system's default definition.

Prepare Move

Specifies whether the downtime during a move or a switch of a resource should be kept as short as possible.

For an application group of Nature MOVE this means that the currently-running group member is not stopped until the replacement member is ready to be started, for example, all parents are available.

For an application group of Nature SERVER this means that a group member will not be stopped until the newly started replacement member is available.

To have the highest possible availability, it is recommended to set this value to YES unless the application cannot support this.

Note: This field is only relevant for application groups with nature MOVE or SERVER. The default value is YES.

Move Mode

Refer to “Move Mode” on page 137 for more details.

Inform List

This field allows you to specify:

- Where the application is registered to
- Where all status changes are propagated to
- Which SMF records are written

Valid receivers are SDF, NMC, IOM, and SMF. You can specify more than one receiver.

The list of values can be blank or comma separated. Neither system automation symbols (AOCCLONEs) nor system symbols are allowed.

Specify NONE to prevent the default from the systems default definition (for application groups of type SYSTEM, see “System Defaults Policy Object” on page 288) or the sysplex default definition (for application groups of type SYSPLEX, see “Sysplex Defaults Policy Object” on page 307) being used.

Owner

Specifies who the operator should contact if there is an error.

Info Link

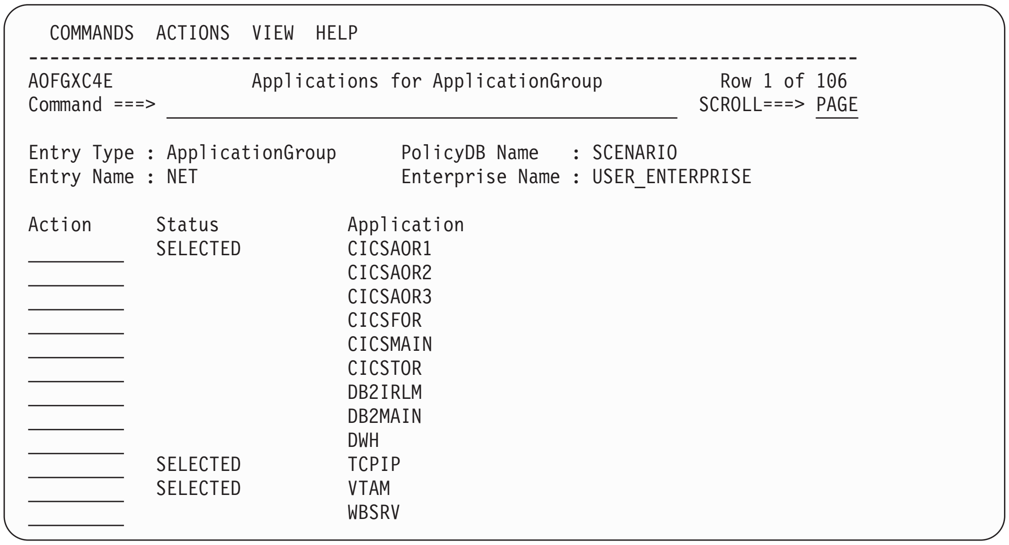
This field can be used to specify a location (for example, a URL) where additional information about the application can be found.

Automatically link Application-Resources

Specifies whether linked applications should also be automatically linked as resources. See also 136.

APPLICATIONS Policy Item

If you select an application group’s policy item APPLICATIONS, you reach the Applications for ApplicationGroup panel. Figure 98 on page 142 shows an example.



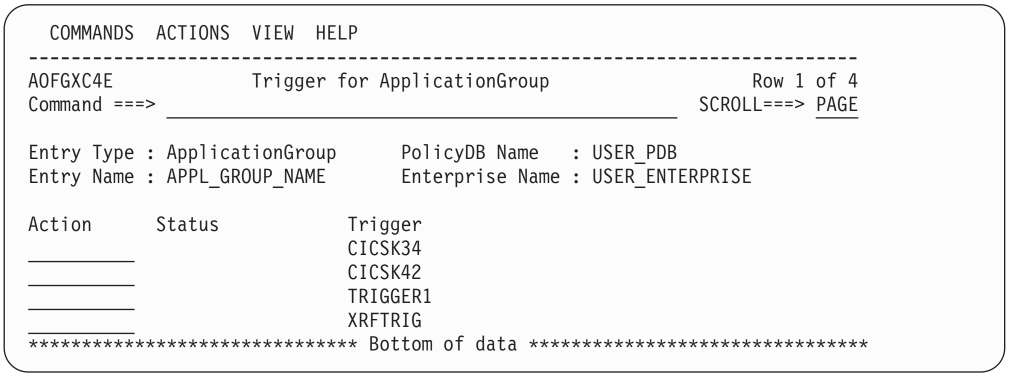
```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E             Applications for ApplicationGroup          Row 1 of 106
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : ApplicationGroup      PolicyDB Name   : SCENARIO
Entry Name : NET                   Enterprise Name : USER_ENTERPRISE

Action      Status           Application
_________   SELECTED         CICSAOR1
_________                    CICSAOR2
_________                    CICSAOR3
_________                    CICSFOR
_________                    CICSMAIN
_________                    CICSTOR
_________                    DB2IRLM
_________                    DB2MAIN
_________                    DWH
_________   SELECTED         TCPIP
_________   SELECTED         VTAM
_________                    WBSRV
```

Figure 98. Applications for ApplicationGroup Panel for System Application Groups

Here you can add or remove applications from an application group. If you add an application to an application group which is already linked to a system, SA z/OS will generate a resource for this application. If you previously set the option *Automatically link Application-Resources* to YES when creating the application group, SA z/OS will also add the new resource into this application group. Otherwise, you simply have an independent application resource that you can start, stop, or automate as required.

TRIGGER Policy Item

If you select an application group's TRIGGER policy item, you reach the Trigger for ApplicationGroup panel. Figure 99 shows an example. For more details about triggers, refer to "Event and Trigger Support" on page 25.



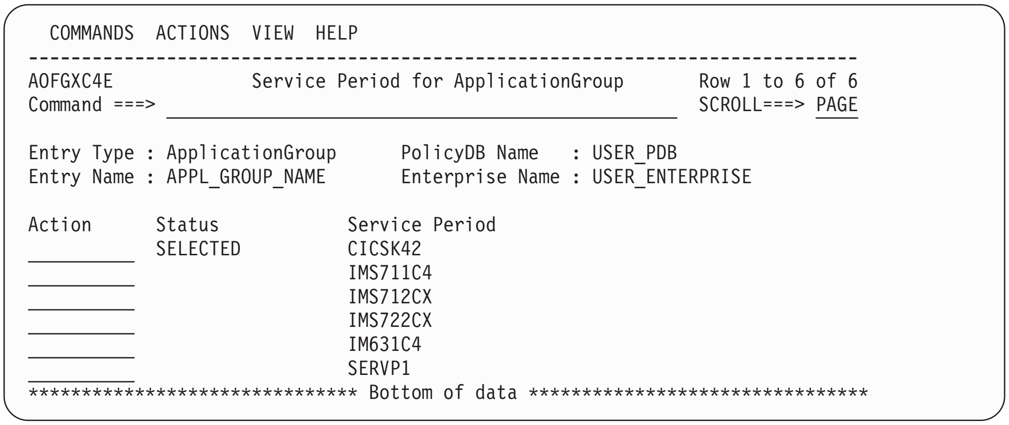
```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E             Trigger for ApplicationGroup                  Row 1 of 4
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : APPL_GROUP_NAME       Enterprise Name : USER_ENTERPRISE

Action      Status           Trigger
__________                   CICSK34
__________                   CICSK42
__________                   TRIGGER1
__________                   XRFTRIG
******************************* Bottom of data ********************************
```

Figure 99. Trigger for ApplicationGroup Panel

Here you can add triggers to or remove them from an application group.

SERVICE PERIOD Policy Item

If you select an application group's SERVICE PERIOD policy item, you reach the *Service Period for ApplicationGroup* panel. Figure 100 on page 143 shows an example.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E            Service Period for ApplicationGroup       Row 1 to 6 of 6
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : APPL_GROUP_NAME       Enterprise Name : USER_ENTERPRISE

Action      Status           Service Period
__________  SELECTED         CICSK42
__________                   IMS711C4
__________                   IMS712CX
__________                   IMS722CX
__________                   IM631C4
__________                   SERVP1
****************************** Bottom of data ********************************
```

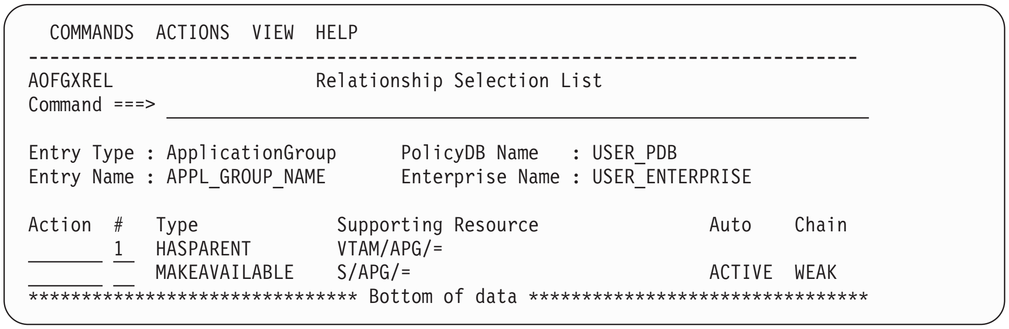
Figure 100. Applications for ApplicationGroup Panel for System Application Groups

Here you can add service periods to or remove them from an application group.

RELATIONSHIPS Policy Item

The Relationship Selection List panel (Figure 101) is displayed if you select policy item RELATIONSHIPS from the Policy Selection panel for application groups. It shows information about existing relationships and allows you to define new relationships.

The ACTION pull-down choice offers actions to edit and delete selected relationships. The **D**elete command immediately deletes the relationship. The **E**dit command leads to the Define Relationship panel (as shown in Figure 101).



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXREL                   Relationship Selection List
Command ===> __________________________________________________________________

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : APPL_GROUP_NAME       Enterprise Name : USER_ENTERPRISE

Action  #   Type            Supporting Resource              Auto    Chain
_______ 1_  HASPARENT       VTAM/APG/=
_______ __  MAKEAVAILABLE   S/APG/=                          ACTIVE  WEAK
****************************** Bottom of data ********************************
```

Figure 101. Relationship Selection List Panel

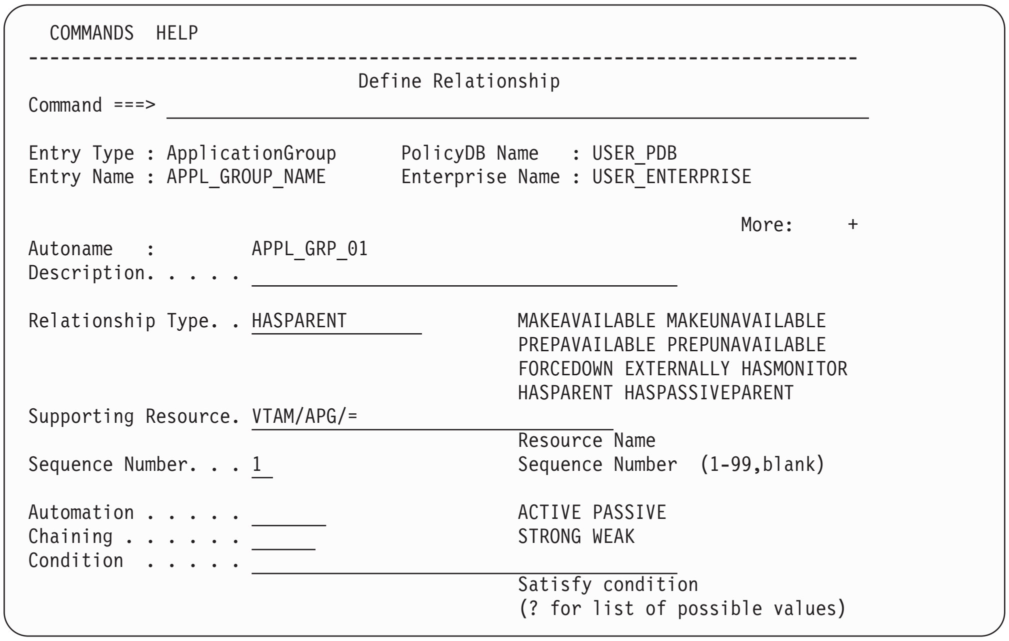
This panel displays information about existing relationships. In all of the relationships for the current resource, this resource is the dependent resource. The columns have the same meaning as described in “Defining Relationships.”

Entering the *NEW* command from this panel to define new relationships takes you to the Define Relationship panel described in “Defining Relationships.”

Defining Relationships

You define relationships between dependent resources and supporting resources. In the customization dialog you always define relationships from the point of view of the dependent resource. So the resource that you are currently editing will become the dependent resource for the relationship dependency being defined. If you select this policy item, in the upcoming Define Relationship panel, (Figure 102 on page 144) you will specify the name of the supporting resource (in field *Supporting*

Resource).



```
  COMMANDS  HELP
------------------------------------------------------------------------------
                               Define Relationship
Command ===>

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : APPL_GROUP_NAME       Enterprise Name : USER_ENTERPRISE

                                                                 More:     +
Autoname   :        APPL_GRP_01
Description. . . . .

Relationship Type. . HASPARENT                 MAKEAVAILABLE MAKEUNAVAILABLE
                                               PREPAVAILABLE PREPUNAVAILABLE
                                               FORCEDOWN EXTERNALLY HASMONITOR
                                               HASPARENT HASPASSIVEPARENT
Supporting Resource. VTAM/APG/=
                                               Resource Name
Sequence Number. . . 1                         Sequence Number  (1-99,blank)

Automation . . . . .                           ACTIVE PASSIVE
Chaining . . . . . .                           STRONG WEAK
Condition  . . . . .
                                               Satisfy condition
                                               (? for list of possible values)
```

Figure 102. Define Relationship Panel

This panel shows either the automation name (*Autoname*) for ApplicationGroups or the subsystem name (*Subsystem*) for Applications.

This panel contains the following fields that can be filled in:

Description
You can enter a forty-character text to describe the relationship.

Relationship Type

Here you define the relationship between the application group, referred to as the dependent resource, with any other resource referred to as the supporting resource. The supporting resource can be an application, application group, or monitor resource.

All relationships and conditions are only observed by automation if the requested goal of the involved resources allow them. Dependencies are only checked if they are of importance for the existing desired goal.

Select one of the following values:

MAKEAVAILABLE
The dependent resource cannot be started until the specified *Condition* has been fulfilled for the supporting resource. So for example, if you specify *WhenAvailable* as the relationship condition, SA z/OS will start the dependent resource as soon as the supporting resource itself is available, if no request or goal contradicts.

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

MAKEUNAVAILABLE
The dependent resource cannot be stopped until the specified *Condition* has been fulfilled for the supporting resource.

PREPAVAILABLE
If the desired goal of the resource is *MAKEAVAILABLE*, the PRESTART commands for the dependent resource are issued as soon as the specified *Condition* has been satisfied for the supporting resource.

PREPUNAVAILABLE
If the desired goal of the resource is *MAKEAUNVAILABLE*, the SHUTINIT commands for the dependent resource are issued as soon as the specified *Condition* has been satisfied for the supporting resource.

FORCEDOWN
To force a shutdown in case the supporting resource has reached the specified satisfactory condition.

EXTERNALLY
The supporting resource will, at certain times, perform various automated actions for the dependent resource. These actions are specified through the relationships condition. For example, with this selection you specify that an external resource is responsible for starting or stopping the current (dependent) resource.

HASMONITOR
For relationships defining connections between application or application group resources and monitor resources.

HASPARENT
With this selection you specify that the current application or application group has a parent. The parent is the supporting resource that you specify in the *Supporting Resource* entry field (see "Supporting Resource" on page 146).

The dependent resource cannot be started until the supporting resource is available.

The supporting resource cannot be stopped before the dependent resource is unavailable.

The supporting resource will, at certain times, perform various automated actions for the dependent resource. These actions are specified through the relationships condition (see "Condition" on page 147). For example, if you specify *StartsMe* as the relationship condition, the supporting resource will start the dependent resource whenever the supporting resource is started.

A request to start the dependent resource results in issuing a start request against the supporting resource if the supporting resource is down.

HASPASSIVEPARENT

The only difference between the HasPassiveParent and HasParent relationship is that a request to start the dependent resource will not result in the issuing of a start request against the supporting resource if the supporting resource is down. A start request must be issued against the supporting resource before the start request for the dependent resource is processed.

Supporting Resource

Here you enter the name of the resource that the dependent resource has the previously defined relationship with. This name consists of two or three parts, separated by slashes, where the last part shows the system name. You can use the equal sign '=' which then stands for the local system of the dependent resource. The syntax for valid formats is similar to the syntax for generated resources:

```
APG-AutomationName/APG            (for sysplex APG)
APG-AutomationName/APG/MVSSysname  (for system  APG)
APG-AutomationName/APG/=           (for system  APG)
SubSystemName/APL/MVSSysname       (for APL)
SubSystemName/APL/=                (for APL)
MonitorName/MTR/MVSSysname         (for MTR)
MonitorName/MTR/=                  (for MTR)
```

Note:

If the supporting resource is an APL resource, a MTR resource, or a system APG resource, one system automation symbol (&AOCCLONE., ~ or &AOCCLONE*x*. with *x*=1 through 9, A through Z) can be specified in the name. System symbols cannot be specified.

For MVSSysname the special notation with a trailing 'plus-sign' is allowed.

Sequence Number

This number is used when the ACF is built. Each relationship which has a number assigned will be put in the ACF PARENT=(...) field. The first subsystem name in the list where the subsystem is defined in the current sysplex is used for AOCQRY. The position within the list is given by this number. Relationships with the same number are sorted within their sequence number. Valid values are 1 - 99 and blank.

Note:

Specifying this number is only valid for APL resources with a local system specification, i.e. subsysname/APL/=. The number is forced to blank for relationships where the supporting resource is an APG , an MTR resource or an APL resource that has not specified the local system.

Automation

Defines the processing of SA z/OS in order to satisfy the relationship:

ACTIVE

Specifies that SA z/OS will attempt to bring the resource to the desired condition to satisfy the dependency. This is the default for MAKEAVAILABLE and MAKEUNAVAILABLE.

PASSIVE

Specifies that SA z/OS will wait until the resource is in the desired condition to satisfy the dependency. That is, the request to bring the resource into the desired condition needs to come from outside, such as an operator command.

This field is forced to blank for HASMONITOR, HASPARENT, HASPASSIVEPARENT, EXTERNALLY, and FORCEDOWN relationships.

This field is forced to PASSIVE for WhenGroupHasNotFailed, WhenHardDown or any WhenObserved... condition.

Chaining

Defines whether the status of the supporting resource, or of the attached subtree is to be considered:

STRONG

Specifies that SA z/OS considers the status of the attached subtree.

WEAK

Specifies that SA z/OS considers just the status of the supporting resource (this is the default for MAKEAVAILABLE and MAKEUNAVAILABLE).

This field is forced to blank for HASMONITOR, HASPARENT, HASPASSIVEPARENT, EXTERNALLY, and FORCEDOWN relationships. This field is forced to WEAK for PREPAVAILABLE and MAKEAVAILABLE relationships with a WhenGroupHasNotFailed condition.

Strong chaining causes SA z/OS to check all relationships of the involved resources along the attached subtree. Weak chaining simply requires that the action is permitted on the dependent resource.

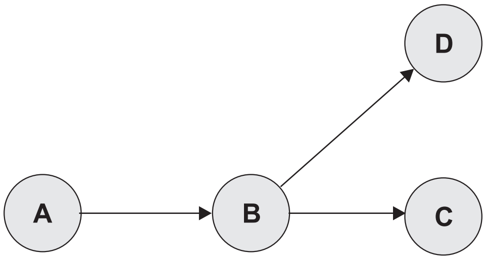


Figure 103. Example for Strong Chaining

In the example shown in Figure 103 for strong chaining, SA z/OS will also consider the dependency between resources B and C and B and D. So A will only be started if B, C and D are up. Or C will only be stopped if A and B are down.

Condition

Defines the condition the supporting resource must have in order to satisfy the relationship. Enter a ’?’ to get to the Relationship Condition Selection panel from where you can select an appropriate value.

Refer to “Relationship Conditions” on page 148 for more information on available condition values and the context that they may be used in. If no value is given for this field the following default values will be set:

blank for HASPARENT

StartsMe

for EXTERNALLY

WhenAvailable

for MAKEAVAILABLE and PREPAVAILABLE

WhenDown

for MAKEUNAVAILABLE and PREPUNAVAILABLE

WhenObservedDown

for FORCEDOWN

This field is forced to blank for HASMONITOR relationships.

Relationship Conditions: The relationship conditions that are available depend on the relationship.

The relationship conditions for HASPARENT, HASPASSIVEPARENT, and EXTERNALLY are:

StartsMe
The dependent resource will be started whenever the supporting resource is started.

StopsMe
The dependent resource will be stopped whenever the supporting resource is stopped.

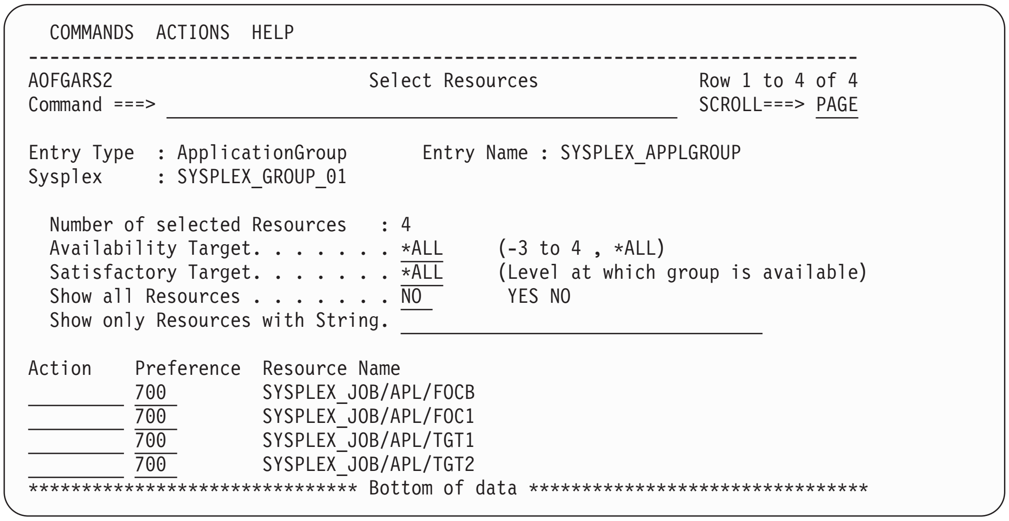
StartsAndStopsMe
The supporting resource will start the dependent resource whenever it is started itself. The supporting resource will stop the dependent resource whenever it is stopped itself.

Refer to the online help for the inventory of available relationship conditions.

RESOURCES Policy Item

When you select policy item RESOURCES, you are first presented with a selection panel if the application group is linked to more than one system or sysplex. There are three different panels depending on whether the application group is of type Server, Move, or Basic. The panel shown in Figure 104 is for an application group of type Server.

If you want to add resource members into an application group of nature MOVE or SERVER, you can attach preference values to these resources (see the *Preference* column in Figure 104).



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGARS2                        Select Resources               Row 1 to 4 of 4
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type  : ApplicationGroup       Entry Name : SYSPLEX_APPLGROUP
Sysplex     : SYSPLEX_GROUP_01

  Number of selected Resources   : 4
  Availability Target. . . . . . . *ALL     (-3 to 4 , *ALL)
  Satisfactory Target. . . . . . . *ALL     (Level at which group is available)
  Show all Resources . . . . . . . NO        YES NO
  Show only Resources with String. __________________________________

Action    Preference  Resource Name
_________ 700         SYSPLEX_JOB/APL/FOCB
_________ 700         SYSPLEX_JOB/APL/FOC1
_________ 700         SYSPLEX_JOB/APL/TGT1
_________ 700         SYSPLEX_JOB/APL/TGT2
******************************* Bottom of data ********************************
```

Figure 104. Select Resources for ApplicationGroup Panel

This panel has the following fields (unless stated, the fields apply to all application group types):

Number of selected Resources
This shows the current number of selected Resources.

Availability Target

This *Availability Target* is different depending on the type of application group:

Server Valid values for Server application groups are:

***ALL** This means that all selected Resources must be *available* before the current application group-derived Resource is considered as *available*.

number

This (positive) *number* is the number of selected Resources that must be *available* before the current application group-derived Resource is considered as *available*. This number must be less than or equal to the value in the *Number of selected Resources* field.

If the value entered is greater, it will be changed to the maximum allowed when you leave the panel.

-*number*

This (negative) *number* is the number of selected Resources *less than* the total that must be *available* before the current application group-derived Resource is considered as *available*. This number must be less than the value in the *Number of selected Resources* field.

If the value entered is less than or equal to the minimum value that is allowed, it will be changed to the minimum when you leave the panel.

Move The *Availability Target* is always 1.

Basic The *Availability Target* is not applicable.

Satisfactory Target

The *Satisfactory Target* is only applicable to Server application groups. It specifies the number of members in the group that must be active before the automation manager treats the group as being in a satisfactory state.

Valid values for server application groups are the same as for the Availability Target.

The Satisfactory Target may be less than, or equal to the Availability Target, but cannot be higher.

If the Satisfactory Target is of type -*number* then it is treated relative to the adjusted Availability Target.

Show all Resources

This can be:

YES

Shows all selectable Resources.

NO

Only shows *selected* Resources.

Show only Resources with String

If you enter a string in this field, only matching Resources are listed. No wildcards are supported.

Action

The following Actions are valid from the ACTIONS menu and Action prompt:

(P)RIMARY
Sets preference value to 700.

SECON(D)ARY
Sets preference value to 400.

(A)DD
Sets preference value to 700.

RE(M)OVE
Removes preference value.

(S)ELECT
Sets preference value to 700.

Preference

A Resource becomes SELECTED when a positive number is entered to this field. The preference value says what ″role″ the selected Resource has. The maximum valid value is 3200. Any value greater than 3200 will be changed to that valid maximum.

The following roles are related to the preference values given:

>2600
The resource always selected regardless of its state. Use this value if the APL or APG resource must run.

≥2000
The resource always selected unless the observed status of the resource is SYSGONE or HARDDOWN. It is only selected when no other viable member exists. A viable member is a resource that can be started and that is allowed to be started. In principle, the observed status is one of the following: SoftDown, Starting, Degraded, Available or WasAvailable.

1000
The resource always started, other resources stopped. (Use if you really want the APL or APG resource to run).

700
The resource is selected for startup after an IPL and after recovery situations. Members with preference 300 will be stopped in favour of this group. It is not stopped unless a very high preference alternative becomes available. This value is the default. Use it in a MOVE group for a resource that normally runs.

600
Used as a threshold. A resource with a preference of 600 and lower will not be selected at group startup.

500
This resource is not selected for startup after IPL but it is started as a backup after failure and continues running even though normal preference (700) alternatives become available. It is stopped at the next (scheduled) outage. Use this value in a MOVE group for the backup resource.

400
This resource is not selected for startup after IPL but it is started as a

backup after failure. The difference to preference 500 is that the resource is stopped when normal preference (700) alternatives become available again.

300
Like 200, but a move to an alternate backup (Preference 500) would only occur at a (scheduled) outage.

200
Resource started as backup, stopped as soon as better alternative (Preference 450 and greater) becomes available. (Use where you can have a backup, but you don't really want one).

1 This means that the member will always be deselected and will always have a MakeUnavailable vote propagated to it. This can be useful if you want complete manual control of the move process.

0 This means that the member is a passive member and will never get a vote propagated to it by the group. If a member has its preference changed so it becomes passive, any votes that have been propagated to it from the group will be withdrawn.

Note: SA z/OS offers the possibility to consider the *workload* in its decision making via resource aware application move. This only takes place if the applications in a system or sysplex have the *same* effective preference values. It also does not affect any preference values that you have defined for your resources. For more details see "Resource Aware Application Move" in *IBM Tivoli System Automation for z/OS User's Guide*.

Preferences other than those listed are accepted. You can find more information on the meaning of preferences in "Preference field description" on page 150 and in "Controlling Application Move with Preference Values" in *IBM Tivoli System Automation for z/OS User's Guide*.

The effective preference of each group member is calculated to determine which group members get picked to be activated by the group. The primary inputs into this are the base preference (as specified through the customization dialogs) and the preference adjustment (specified indirectly through the INGGROUP interface). These are added together to produce the resource's adjusted preference (this is the same as the preference value you actually enter the INGGROUP panels).

The following modifiers are then applied to the resource's adjusted preference if it is >1.

Bonus	Reason
25	Member is currently selected and has an adjusted preference ≤1000.
175	Member is currently selected and has an adjusted preference >1000.
220	Member has observed status of Starting or Degraded and the system is not being stopped.
225	Member has observed status of Available or Stopping and the system is not being stopped.
–400	Member has an adjusted preference >1500 and either the system is being stopped or the system has already been stopped and the member thus has observed status of SysGone.

The working value for the preference may then be set to an absolute value if any of the following conditions are satisfied:

Value	Condition
3600	The member has an adjusted preference of 0 (indicating a passive member) and it has an observed status of Starting, Degraded, Available or WasAvailable.
100	The member has a working preference between 2–2599 and requires a resource on an Avoided system.
1	The member has: • An adjusted preference of 0 (indicating a passive member) and its observed status is none of Starting, Degraded, Available or WasAvailable. • A working preference of 2000–2599, there is a viable alternative resource and the member either has an observed status of HardDown or SysGone or the system that it is on is being stopped. • A working preference <2000 and has observed status of HardDown or SysGone. • A working preference <2600 and has observed status of StandBy, Unknown or Problem. • A working preference <2600, has observed status of Stopping or SoftDown and its automation flag is turned off. • A working preference <2600 and requires a member on an excluded system.

After all of these adjustments have been made, this produces the member's effective preference. Normally a Move or Server group will pick the members with the highest effective preferences to be available, however there are three special rules to take into consideration:

1. A member with an effective preference of 1 may never be picked. SA z/OS sends a MakeUnavailable vote.
2. A member with an effective preference <600 may only be picked if the group is in recovery mode (that is, it is recovering from a failure).
3. A member requiring a resource on an excluded system must be sent a MakeUnavailable vote, even if it has been picked. This can occur with resources that have adjusted preferences >2600.

Various scenarios are described in Table 6.

Table 6. Preference Value Scenarios

Scenario	Description
Absolute manual control	Member to be activated gets preference 3200, all others get preference 1. The member with the 3200 preference will be selected. All others will always have a MakeUnavailable vote propagated to them. This does not change even if the system with the selected member leaves the sysplex. A shutdown of the primary system will not initiate a move of the application, as the 400 point penalty is not sufficient to drop its preference below 2600.

Table 6. Preference Value Scenarios (continued)

Scenario	Description
Move only after system failure	Primary member defined preference 2800, backups preference 500. As long as the primary is not SYSGONE it will be selected. If the primary is SYSGONE, its preference drops to 2575 (2800 − 400 + 175), allowing an alternative viable member to be selected. The backups will only be viable if the resource was previously active because of the groups recovery mode. A shutdown of the primary system will not initiate a move of the application. Although the primary systems preference will be below 2600, there will be no viable alternatives as the backups may only be used if the group is in recovery mode.
Move only if system down or stopping	Primary member defined preference 2800, backups preference 700. As long as the primary member is not SYSGONE and its system is not being stopped it will be selected. If the primary member is SYSGONE or its system is being stopped, its preference drops to 2575 (2800 − 400 + 175), allowing an alternative viable resource to be selected. The backups are always viable. A shutdown of the primary system will initiate a move of the application during the shutdown.
Sticky move only after system down or stopping	Primary member defined preference 2801, backups preference 2800. As long as the primary member is not SYSGONE and its system is not being stopped, it will be selected. If the primary becomes SYSGONE or its system is being stopped, its preference drops to 2576 (2801 − 400 + 175), allowing one of the backups to be selected. The sticky bonus raises its preference to 2975 (2800 + 175), meaning that even when the primary returns the selected backup system will remain selected. It will only be deselected if it goes to SYSGONE or its system is being stopped, in which case the primary will be re-selected if it is available otherwise another backup would be picked. A shutdown of the primary system will initiate a move of the application during the shutdown.

Table 6. Preference Value Scenarios (continued)

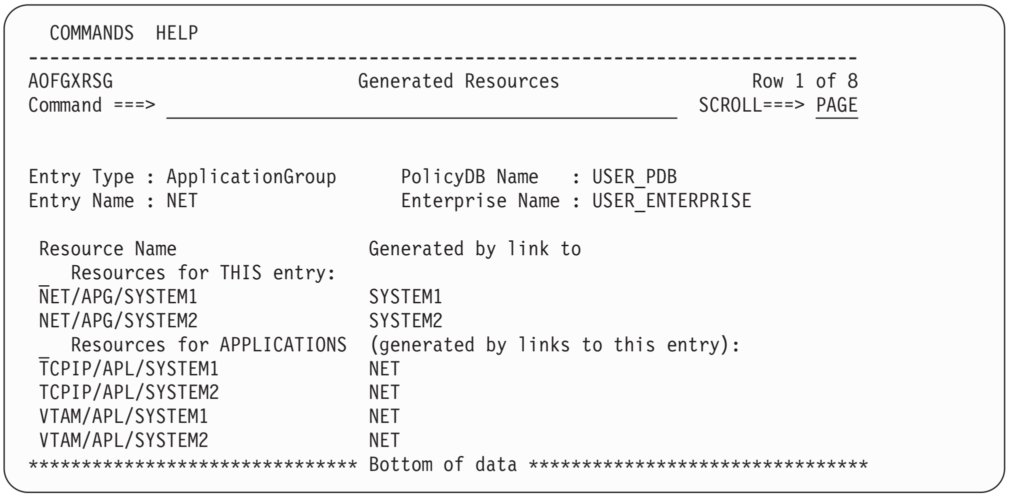
Scenario	Description
Move only after system or application failure	Primary member defined with preference 2400, backups preference 500. As long as the primary member does not fail to SYSGONE or HARDDOWN it remains selected. If it gets to HARDDOWN or SYSGONE without failing, there are no viable alternatives, so it will not be moved. If it does fail, its preference is between 2600 and 2000 and the backups are viable, so one of them can be picked. The backups are only viable if the group is in recovery mode (that is, the member was active when the failure occurred).
Move only if system down, being stopped or application failure	Primary member defined with preference 2400, backups preference 700. As long as the primary member is not SYSGONE or HARDDOWN and its system is not being stopped it remains selected. If it is HARDDOWN or SYSGONE or its system is being stopped its preference is between 2600 and 2000, so a viable alternative can be selected. A shutdown of the primary system will initiate a move of the application, as it will have lost its 225 point bonus for being available and have incurred a 400 point penalty, bringing its preference down to well below 2600, allowing the selection of an alternative resource if there is a viable one.
Sticky move if system downor being stopped, move after application failure	Primary member defined with preference 2401, backups preference 2400. As long as the primary is not SYSGONE or HARDDOWN and its system is not being stopped it remains selected. If it is SYSGONE or HARDDOWN or its system is being stopped its preference is between 2000–2600, so a viable alternative may be chosen. The sticky bonus raises the preference of the selected member to 2575 (2400 + 175), so it will remain selected until it gets to a state of SYSGONE or HARDDOWN or its system is being stopped, at which point another viable alternative will be selected, with the primary member having preference. The individual shutdown of the active member will not trigger a move. A shutdown of the primary system will result in a move of the application. It will have a 400 point penalty, dropping its preference to between 2600 and 2000, permitting the selection of a viable alternative.

Table 6. Preference Value Scenarios (continued)

Scenario	Description
Sticky move group, unless member stopped	Primary member defined with preference 701, backups preference 700. As long as the primary member is startable, it remains selected. If it becomes non-startable, another member is selected and that remains selected as long as it is startable. Individually stopping the active member will trigger a move. If the members were defined with preferences >1500, the bonus would be 175 points rather than 25 points.
Sticky move group, until told otherwise	Primary member defined with preference 3200, backups with preference 1600. With base preferences of 3200, you get it picking a member when the sysplex is IPLd and retaining that one until an operator uses INGGROUP to make it pick another one. Beware of timing problems as the first system with a member to register will probably get it.

GENERATED RESOURCES Policy Item

The Generated Resources panel appears if you select policy item *GENERATED RESOURCES* from the Policy Selection panel for ApplicationGroups. This panel shows all resources that are generated by SA z/OS for the application group itself and all the resources for the applications that are members of this application group. This panel is for information only. No actions are allowed.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGXRSG                        Generated Resources                 Row 1 of 8
Command ===> ________________________________________________ SCROLL===> PAGE


Entry Type : ApplicationGroup       PolicyDB Name   : USER_PDB
Entry Name : NET                    Enterprise Name : USER_ENTERPRISE

 Resource Name                  Generated by link to
 _   Resources for THIS entry:
 NET/APG/SYSTEM1                SYSTEM1
 NET/APG/SYSTEM2                SYSTEM2
 _   Resources for APPLICATIONS  (generated by links to this entry):
 TCPIP/APL/SYSTEM1              NET
 TCPIP/APL/SYSTEM2              NET
 VTAM/APL/SYSTEM1               NET
 VTAM/APL/SYSTEM2               NET
******************************* Bottom of data ********************************
```

Figure 105. Generated Resources Panel

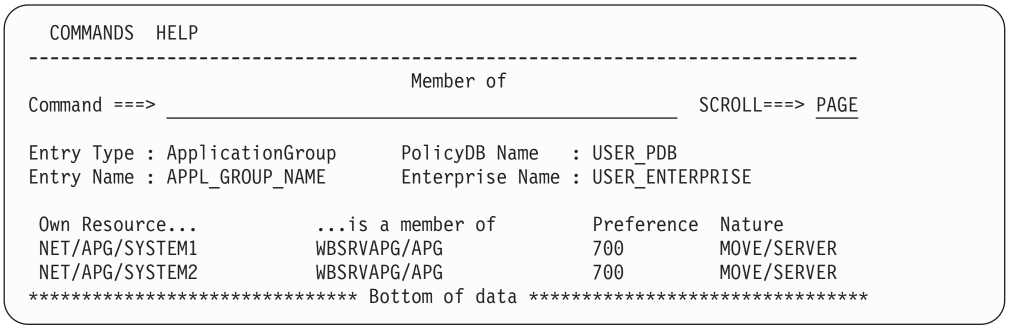
This panel displays the following information:

Resources for THIS entry
Shows the resources generated for the application group. The display is different for application groups of type SYSPLEX or SYSTEM: No system information is shown for Sysplex Application Groups, as several systems can be part of the sysplex.

Resources for APPLICATIONS (generated by links to this entry)
Shows the resources generated for applications that are linked to the application group.

MEMBER OF Policy Item

The Member of panel appears if you select policy item *MEMBER OF* from the Policy Selection panel for application groups.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
                                    Member of
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : ApplicationGroup      PolicyDB Name   : USER_PDB
Entry Name : APPL_GROUP_NAME       Enterprise Name : USER_ENTERPRISE

 Own Resource...           ...is a member of          Preference  Nature
 NET/APG/SYSTEM1           WBSRVAPG/APG               700         MOVE/SERVER
 NET/APG/SYSTEM2           WBSRVAPG/APG               700         MOVE/SERVER
******************************* Bottom of data ********************************
```

Figure 106. Member of Panel for ApplicationGroups

This panel lists all Resources, where this ApplicationGroup is a member ″resource-wise″; that is, it lists all relations from Resources that have been derived from the current ApplicationGroup to other Resources. This panel is for information only. No data can be changed.

A ″Member of″ relation can be added or removed in the RESOURCES policy of the entry in the list’s ″member of″ column (that is in the RESOURCES policy of the APG Entry Name, that the ″member of″ Resource was derived from).

If the owning (″is a member of″) APG resource is of nature SERVER or MOVE, column *Preference* shows the preference value that was set for the current resource in the RESOURCES policy item of the owning resource. If the owning resource’s nature is BASIC, it shows ″SELECTED″. For more details about available preference values see “Creating a New ApplicationGroup” on page 134.

In our example from Figure 106, the resource *NET/APG/SYSTEM1* in turn is a member of resource *WBSRVAPG/APG*.

Application Policy Object

An *application* is a z/OS subsystem, started task, application, batch job, or non-MVS resource that runs on a system in the SA z/OS enterprise. SA z/OS monitors applications through the messages they issue to the operator. The term application is also used to refer to a subsystem or process on an SNMP-capable machine. Note that SA z/OS does not provide automation for applications on non-MVS systems.

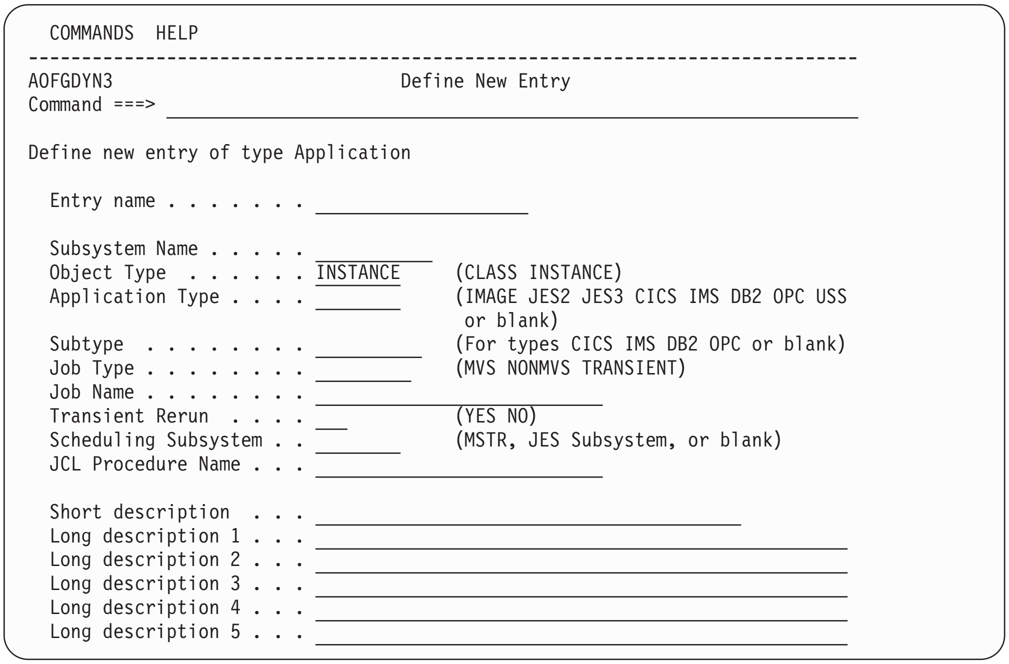
An application cannot be linked to a system directly. It must be defined as part of an application group, and it may belong to more than one application group. If a system is assigned multiple application groups containing the same application, SA z/OS ensures that the application is defined only once in the automation control file.

The SA z/OS customization dialog allows concurrent user access capability for the Application entry type. See Appendix A, “Concurrent Customization by Multiple Users,” on page 387 for further information.

Creating a New Application

How to create new policy objects in general is described in “How to Create New Policy Objects” on page 37.

Figure 107 shows the panel that is displayed when you create a new application from the Entry Name Selection panel for applications.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN3                           Define New Entry
Command ===> _________________________________________________________________

Define new entry of type Application

  Entry name . . . . . . . ____________________

  Subsystem Name . . . . . ___________
  Object Type  . . . . . . INSTANCE     (CLASS INSTANCE)
  Application Type . . . . ________     (IMAGE JES2 JES3 CICS IMS DB2 OPC USS
                                         or blank)
  Subtype  . . . . . . . . __________   (For types CICS IMS DB2 OPC or blank)
  Job Type . . . . . . . . _________    (MVS NONMVS TRANSIENT)
  Job Name . . . . . . . . ___________________________
  Transient Rerun  . . . . ___          (YES NO)
  Scheduling Subsystem . . ________     (MSTR, JES Subsystem, or blank)
  JCL Procedure Name . . . ___________________________

  Short description  . . . ________________________________________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 107. Define New Entry Panel for Applications

When you define a new application, you provide the following information:

Entry Name
: This field is required. It assigns a unique twenty-character name to the application you are defining. It is often convenient to use the same name for both the subsystem name and the application name. In some cases, however, it may be useful to have multiple applications with the same subsystem name. This enables you to have a standard ″tree″ of subsystems but to assign the right characteristics of an application to a specific system. The first character of the application name may not be numeric.

Subsystem Name
: The subsystem name is required. It defaults to the application’s entry name if the entry name complies with the restrictions for subsystem names (see below). Press Enter to have the default displayed in the field.

The subsystem name provides a unique, eleven-character name for the Application that you are defining. This name is used by SA z/OS automation. You can have multiple application instances with the same subsystem name, however, you should take care to ensure that only one of them is linked to any given system. The subsystem name of an application class must be unique across the whole policy database, that is, it cannot be the same as the subsystem name of any other application instance or class.

Note that the subsystem name of an existing application can be modified by overtyping the **Subsystem Name** field in the Application Information policy panel.

The restrictions on the subsystem name are:

- The length is limited to eleven characters.
- Valid characters are any alphanumeric character or any of @, #, $, ? or _ (underscore), with the first character being alphanumeric or one of @, #, $ or ?.
- Refer to the online help for a list of values that are reserved for SA z/OS and cannot be used.

When defining applications on SNMP target systems, it is recommended that the subsystem name be the same as the name of the process or subsystem on the SNMP target system.

For applications of type IMAGE, the same value is required for the subsystem and the job name. Therefore the subsystem name must comply with the restrictions for job names (see "Job Name" on page 160).

Object Type

CLASS
If the entry represents common policy that is to be inherited by one or more other application entries that are defined as INSTANCEs, select this object type. CLASSes are used only to define inherited policy; they do not represent actual applications that are managed by SA z/OS.

INSTANCE
If the entry represents an application that is to be managed by SA z/OS, select this object type. If you link an INSTANCE to a CLASS you can omit policy items from the INSTANCE, which allows that policy to be inherited from the specified CLASS.

Use of the CLASS object type allows you to define a common policy for multiple applications. This is advisable if these applications share common characteristics and functions. It decreases effort during:

- Initial definition
- Administrative changes (using the customization dialog)
- Run time changes (using the SA z/OS automation engine)

An application instance can be linked to a class using the UPWARD CLASS policy item. If a policy item is not specified for the instance, it is inherited from the class. Conversely, a policy item that is specified for an instance overrides any class specification.

Flag definitions for classes or defaults cannot be merged with specific resource definitions. For example, specifying a RESTART flag with an exit at class level and a RESTART flag with a NOAUTO period at resource level will not result in a resource definition with both. The presence of any definition at resource level takes precedence over those at class and default level. Likewise a class definition takes precedence over a default. The definitions for a specific flag type are not merged.

The inheritance of policy from an application class occurs during SA z/OS initialization. If an application policy item is not specified in either the application instance or the class, defaults are used from the application defaults, system defaults, and environment setup objects. Some policy items are required to be specified for an application instance. Except for job name, these same items need to be specified for an application class, but note they will never be inherited.

SA z/OS supports run time changes to an application class. If the ACF command is used to change a policy item for a class, all those instances belonging to the class and inheriting that policy item are also updated.

Application Type

The application type is not required. You may specify one of the following values:

- JES2 if the application represents a JES2 application, see "Policy Items for JES2 Subsystems" on page 212.
- JES3 if the application represents a JES3 application, see"Policy Items for JES3 Subsystems" on page 216.
- DB2 if the application represents a DB2 application, see"Policy Items for DB2 Subsystems" on page 220.
- CICS if the application represents a CICS application, see"Policy Items for CICS Subsystems" on page 218.
- IMS if the application represents an IMS application, see"Policy Items for IMS Subsystems" on page 223.
- OPC if the application represents a TWS application, see"Policy Items for OPC Subsystems" on page 227.
- USS if the application represents an USS application, see"Policy Items for USS Subsystems" on page 229.
- IMAGE if the application represents a system image (to enable enhanced Parallel Sysplex functions). For details, see *IBM Tivoli System Automation for z/OS Customizing and Programming*.

If you specify one of these application types, additional policy options specific to their automation are displayed on the SA z/OS Policy Selection panel. Note that once you have specified the value and created the Application entry, the value cannot be changed. Only the blank value can be changed afterwards.

Subtype

Specifies the region subtype for CICS, DB2 and IMS or the type for OPC applications. The following values may be specified:

- **CICS:** AOR, TOR, FOR, DOR, QOR, CMAS
- **DB2:** MSTR, SPAS, IRLM, DBM1, DIST, WLMS
- **IMS:** CTL, TP, DBRC, DLS, FP, BMP
- **OPC:** CONTROLLER, TRACKER, SERVER, DATASERVER

For applications without a type specification, a subtype (including blank) can also be specified.

You must specify the subtype for application classes of type DB2 or IMS.

For other applications the value cannot be specified.

Job Type

This can be one of:

MVS This is the standard MVS job type that SA z/OS normally handles. It is typically monitored using the ASCB checker. This job type can be specified explicitly with JOBTYPE=MVS for an application, however, it is the assumed default if the Job Type entry is omitted from the definition of the application.

NONMVS

This is a type of resource that may be started by SA z/OS but which cannot be monitored using an ASCB check or the MVS D A method.

If the AOFATMON routine cannot be used to monitor this type of resource, either a user-provided application monitor routine is required or the message automation table must be correctly set up to maintain the job status accurately. This status is the only way of monitoring this class of job. NONMVS jobs must have STARTUP and SHUTDOWN commands defined.

TRANSIENT

This is a type of job that may be started by SA z/OS but terminates without the intervention of SA z/OS. It is a short running job that will generally, but not necessarily, terminate before SA z/OS does. It can be monitored using an ASCB check.

As far as monitoring is concerned, a TRANSIENT job is treated as a z/OS job, but it has special properties: transient jobs must START and END before their dependent applications are started.

You can specify shutdown passes for transient jobs if there is a possibility that they may need to be shut down by SA z/OS. These passes are optional.

Job Name

The job name is required. It defaults to the subsystem name if this complies with the restrictions on job names. Press Enter to have the default displayed in the field. For applications of type IMAGE, the job name is *forced* to the subsystem name. Thus, for type IMAGE, the subsystem name *must* comply with the restrictions on job names.

The job name gives a unique name to the application to be run. There is a one-to-one mapping between the MVS job name and the subsystem name. SA z/OS automation interprets messages from MVS by taking the MVS job name and mapping it to the subsystem name.

You can have multiple applications with the same job name but you must ensure that only one of them is linked to a particular system. Duplicates are ignored by the BUILD function.

The restrictions on the job name are:

- It can be up to eight alphanumeric and national characters, including @, #, and $, after clone resolution, provided the first character is not numeric.
- If the job type is NONMVS, you can also use _ and ?.
- You can include system symbols and system automation symbols.
- The special value *MASTER* is allowed for applications of type blank. If the object type is CLASS, a job name is not allowed.

For details about creating the start command, see “JCL Procedure Name” on page 161.

Transient Rerun

This field specifies whether a TRANSIENT application may be rerun within the lifetime of z/OS. The Transient Rerun option involves the concept of a *one-shot transient job*.

YES This indicates that the job will be rerun as appropriate. SETSTATE can be used to set it to RESTART status, and it can be part of a shutdown request with the RESTART option set to YES.

NO This indicates that the job will only be run once in the lifecycle of a particular z/OS. A value of NO affects the processing of the SETSTATE command. NO is the default value for transient jobs.

> **Note:**
>
> On an initial automation policy load, such as at IPL, a one-shot transient will run, terminate and go to ENDED status. If a SETSTATE or INGREQ with restart is attempted it will be rejected because a one-shot transient is only allowed to run once in the lifecycle of a particular z/OS.
>
> A one-shot job remains in an ENDED state across a SA z/OS recycle.
>
> There are some situations when even one-shot transient jobs will be rerun:
>
> - If the automation status file is deleted and NetView is recycled, a one-shot transient job will be restarted.
> - If a job is deleted from the automation policy, status information for that job is deleted during a refresh of the policy. If a later refresh is used to redefine the application to automation, the application status is set to DOWN. This may also result in the restart of a one-shot transient job being restarted.

Scheduling Subsystem

This is optional. It is the name of the subsystem that is to schedule this application, as follows:

MSTR The application is scheduled directly by the MVS/ESA master scheduler.

blank The application is scheduled by the primary JES.

other The application is scheduled by another scheduling subsystem. The value entered is the name of that subsystem. The subsystem specified must be defined as being type JES2 or JES3.

For applications of type IMAGE the value is forced to blank.

JCL Procedure Name

The JCL Procedure Name is optional. This specifies the name of the JCL procedure used to create this instance of the application. If specified, the default start command for the application will be:

```
MVS S procname,JOBNAME=jobname
```

where *procname* is the JCL Procedure Name and *jobname* is the Job Name specified above. If no JCL Procedure Name is specified, the default start command is:

```
MVS S jobname
```

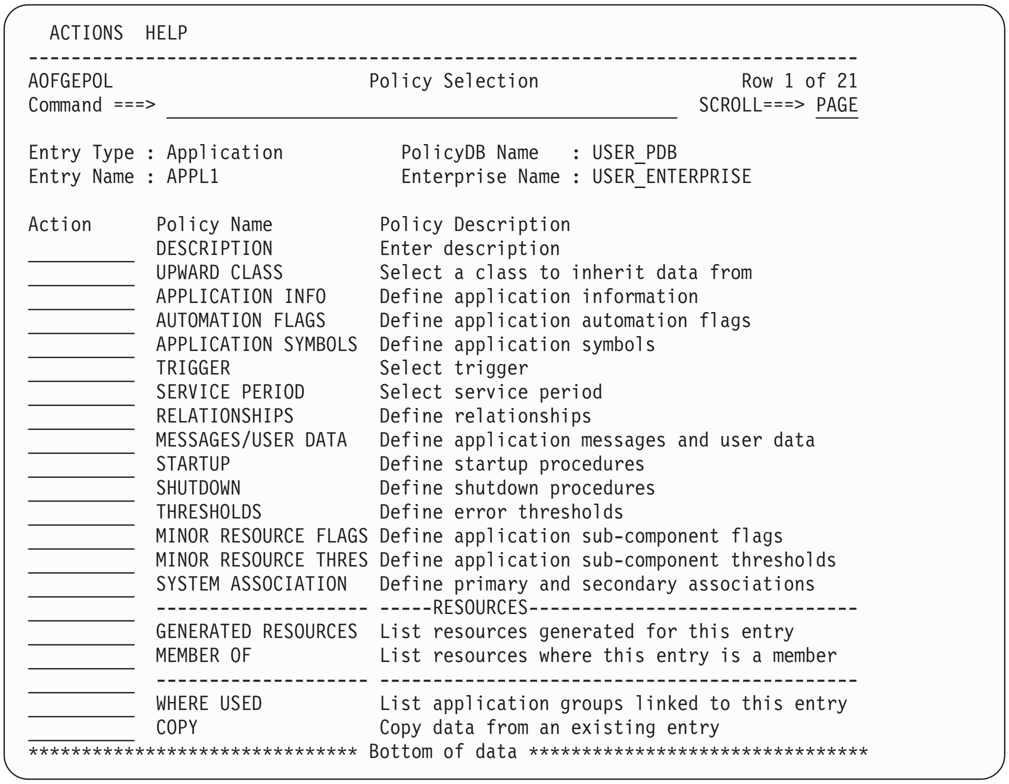
You may include system symbols and system automation symbols (see “Assigning System Automation Symbols (AOCCLONE)” on page 27).

For applications of type IMAGE the value is forced to blank.

Policy Items for Applications

Figure 108 shows the policy items that are displayed when an Application is selected from the Entry Name Selection panel.

Some of the policy items may be set for all application automation on a certain system via the application or system defaults. For example, Thresholds can be set to a default value for all applications. If the definition of an application has no value specified for Thresholds, the value specified in the Application Defaults is used. If no value is specified in the Application Defaults any value specified in System Defaults is used. See also “Application Defaults Policy Object” on page 290 and “System Defaults Policy Object” on page 288.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                  Row 1 of 21
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            DESCRIPTION          Enter description
__________  UPWARD CLASS         Select a class to inherit data from
__________  APPLICATION INFO     Define application information
__________  AUTOMATION FLAGS     Define application automation flags
__________  APPLICATION SYMBOLS  Define application symbols
__________  TRIGGER              Select trigger
__________  SERVICE PERIOD       Select service period
__________  RELATIONSHIPS        Define relationships
__________  MESSAGES/USER DATA   Define application messages and user data
__________  STARTUP              Define startup procedures
__________  SHUTDOWN             Define shutdown procedures
__________  THRESHOLDS           Define error thresholds
__________  MINOR RESOURCE FLAGS Define application sub-component flags
__________  MINOR RESOURCE THRES Define application sub-component thresholds
__________  SYSTEM ASSOCIATION   Define primary and secondary associations
__________  -------------------- -----RESOURCES-------------------------------
__________  GENERATED RESOURCES  List resources generated for this entry
__________  MEMBER OF            List resources where this entry is a member
__________  -------------------- --------------------------------------------
__________  WHERE USED           List application groups linked to this entry
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

Figure 108. Policy Selection Panel for Applications

The unique policy items for Applications are:

UPWARD CLASS
This field is displayed if the application was defined as an INSTANCE or CLASS (refer to “Creating a New Application” on page 157). It lists all entries that have been defined as resource classes and allows you to select one class that this application should be linked to so that it inherits the policy values of that class or class chain.

See “UPWARD CLASS Policy Item” on page 166 for more information.

DOWNWARD CLASS/INST
This policy item is displayed if the application was defined as a CLASS

(refer to “Creating a New Application” on page 157) and it lists all class or instance entries. You can add or remove classes or instances linked to this class.

See “DOWNWARD CLASS/INST Policy Item” on page 166 for more information.

APPLICATION INFO

This policy item displays the Application Information panel that shows the policy for the current application as entered when it was created. It is also used to define the application to SA z/OS automation. You can edit part of the policy information on this panel. The fields that contain policy information that you must not modify are read-only. To change read-only information, you must delete and recreate the application.

See “APPLICATION INFO Policy Item” on page 168 for more information.

AUTOMATION FLAGS

Automation flags allow you to enable and disable automation. You can disable automation for specific time periods or turn it off entirely. You can also enable or disable automation for specific automation phases, such as Initstart, Start, Recovery, Shutdown, and Restart, and for responses to specific messages and status changes within these phases.

Determine whether the automation flags currently set for your system at the system wide defaults and application levels are appropriate for this application. If they are not, define one or more application automation flags in either the application instance or class policy object.

For more information, see “AUTOMATION FLAGS Policy Item” on page 176 and “Automation Flags” on page 20.

APPLICATION SYMBOLS

Application symbols allow you to enter or update application symbols that can be set to various application-specific values. These can then be substituted using the runtime variables SUBSSYMBOL1 through SUBSSYMBOL9. Those variables can be used in a similar fashion to runtime variables like SUBSSPARM, SUBSSCHEDSS, SUBSJOB, etc.

For more information, see “APPLICATION SYMBOLS Policy Item” on page 177.

TRIGGER

Each application may be linked to one trigger. This policy item allows you to link the current application to a trigger. Only one trigger can be selected.

Triggers are used to control the starting and stopping of applications in a single system or a Parallel Sysplex. They determine the behavior of the requested action. For an application to be started, its desired status must be AVAILABLE and its startup trigger must be satisfied.

For an application to be stopped, its desired status must be UNAVAILABLE and its shutdown trigger must be satisfied.

For more information on how to use triggers, see “TRIGGER Policy Item” on page 178 and “Event and Trigger Support” on page 25.

SERVICE PERIOD

Service periods allow users to schedule the availability of applications, application groups, or monitor resources. With this policy item you can link a service period to the application so that it is automatically started up or shut down in the time frame specified with the service period.

For more information, see "SERVICE PERIOD Policy Item" on page 179.

RELATIONSHIPS

Relationships are used to define dependencies between resources that are used to perform in a controlled way startup or shutdown actions of dependent and supporting applications, application groups, and monitor resources.

This policy item has the same functionality for entry types Application, ApplicationGroup, and Monitor Resource. For each application, application group, or monitor resource you may specify several relationships. If you select this policy item, in the upcoming Relationship Selection List panel, you can view existing or define new relationships for the current application, application group, or monitor resource.

You define relationships between dependent resources and supporting resources. In the customization dialog you define relationships always from the point of view of the dependent resource. So the resource that you have currently selected will become the dependent resource, when you define a new relationship.

See "RELATIONSHIPS Policy Item" on page 143 for more information on viewing or defining dependency relationships.

MESSAGES/USER DATA

This policy item allows you to specify automated actions, such as the issuing of a command or reply, to be taken in response to a message issued by the application.

You can also specify some of the application statuses as messages to trigger the command of your choice.

How to proceed in specifying this policy item is described in "MESSAGES/USER DATA Policy Item" on page 180.

STARTUP

This policy item allows you to add, remove or modify startup procedures for the current application.

See "STARTUP Policy Item" on page 195 for more information how to specify the startup automation policy.

SHUTDOWN

You must specify NORMAL shutdown commands for any application that you want to automate with SA z/OS. You may also specify INIT, IMMED, FORCE, and FINAL shutdown commands.

Specifying shutdown commands or replies for the application allows the automation to perform application shutdowns. You can specify different shutdown commands or replies to be issued for the different types of shutdown situations.

How to achieve this is described in "SHUTDOWN Policy Item" on page 199.

THRESHOLDS

Error threshold definitions for an application define how many abend errors can occur before a message is logged or restart processing is stopped. If no error thresholds are set here, the application uses the thresholds defined in the Application Defaults policy object.

Determine whether existing default threshold values are appropriate for this application. If they are not appropriate, create a unique threshold entry for the application.

For more information on thresholds, see "How SA z/OS Uses Error Thresholds" on page 17.

MINOR RESOURCE FLAGS

This policy item allows you to define flags for minor resources for the application. Minor resources are any resources specified to be subsidiary to this application, for example, messages and statuses.

The main purposes in defining minor resources are to let you:

- Trigger automation for minor resources
- Override the automation settings of major resources for a single minor resource, or a select number of minor resources

See the description of the MINOR RESOURCE FLAGS policy item in "MVS Component Policy Object" on page 282 for information on how to manage minor resources and "MINOR RESOURCE FLAGS Policy Item" on page 208.

The automation flags that are associated with minor resources are sometimes called extended automation flags. See "Automation Flags and Minor Resource Flags" on page 25 for information about them.

MINOR RESOURCE THRES

This panel allows you to manage thresholds for minor resources. Minor resources are any resources specified to be subsidiary to this application, for example, messages and statuses.

See "MINOR RESOURCE THRES Policy Item" on page 209 for more details.

SYSTEM ASSOCIATION

This policy item allows you to define primary and secondary systems for the application. Primary and secondary categorizations help SA z/OS determine whether an application should be started by SA z/OS when SA z/OS is initialized on a system. These categories allow SA z/OS to perform automation, such as monitoring or shut down, on applications that can be moved by an external agent.

For a detailed description on how to define system categories with this option see "SYSTEM ASSOCIATION Policy Item" on page 210.

GENERATED RESOURCES

This policy item lists all resources that were generated for the current application. Application resources are generated when a link between an application and a system occurs. Applications can never be linked directly to systems. This link happens indirectly by linking an application to an application group (either in an application's WHERE USED policy item or in an application group's APPLICATIONS policy item.)

The GENERATED RESOURCES panel is displayed, which looks similar to the one shown in Figure 105 on page 155, except that the section, **Resources for THIS entry**, showing the application group resource is missing.

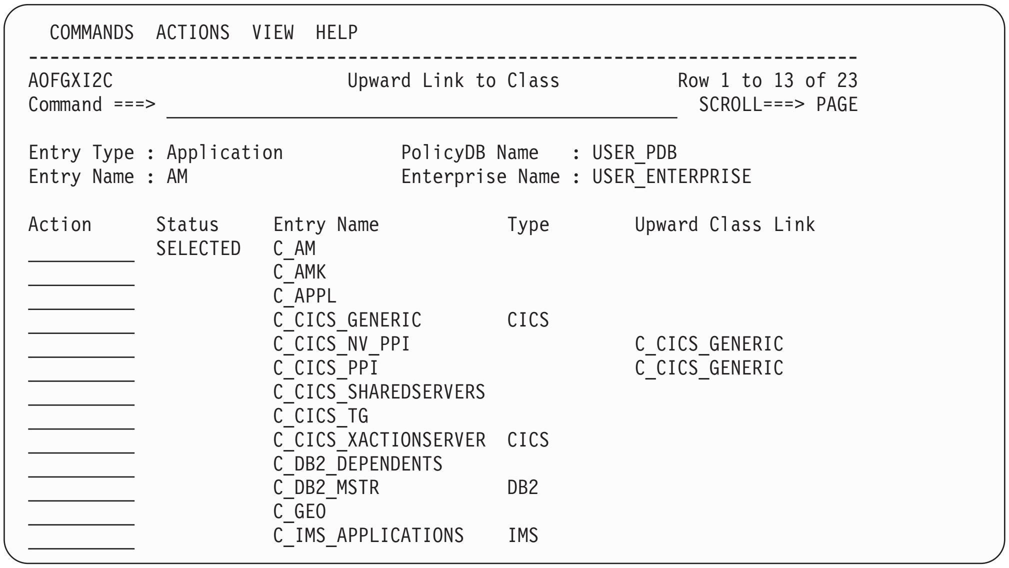
MEMBER OF

This policy item lets you see the APG Resources that the current application is a member of.

See “MEMBER OF Policy Item” on page 156 for more details.

UPWARD CLASS Policy Item

If you select the UPWARD CLASS policy item from the Policy Selection panel for Applications, the Upward Link to Class panel is displayed, as shown in Figure 109.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXI2C                      Upward Link to Class          Row 1 to 13 of 23
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : AM                   Enterprise Name : USER_ENTERPRISE

Action      Status     Entry Name            Type        Upward Class Link
__________  SELECTED   C_AM
__________             C_AMK
__________             C_APPL
__________             C_CICS_GENERIC        CICS
__________             C_CICS_NV_PPI                     C_CICS_GENERIC
__________             C_CICS_PPI                        C_CICS_GENERIC
__________             C_CICS_SHAREDSERVERS
__________             C_CICS_TG
__________             C_CICS_XACTIONSERVER  CICS
__________             C_DB2_DEPENDENTS
__________             C_DB2_MSTR            DB2
__________             C_GEO
__________             C_IMS_APPLICATIONS    IMS
```

Figure 109. Upward Link to Class Panel

This panel allows you to manage the upward link from an application class or instance to a class of the same type. This link allows the instances or classes to inherit any policy data that has been defined in the selected class or other classes further up in the class chain.

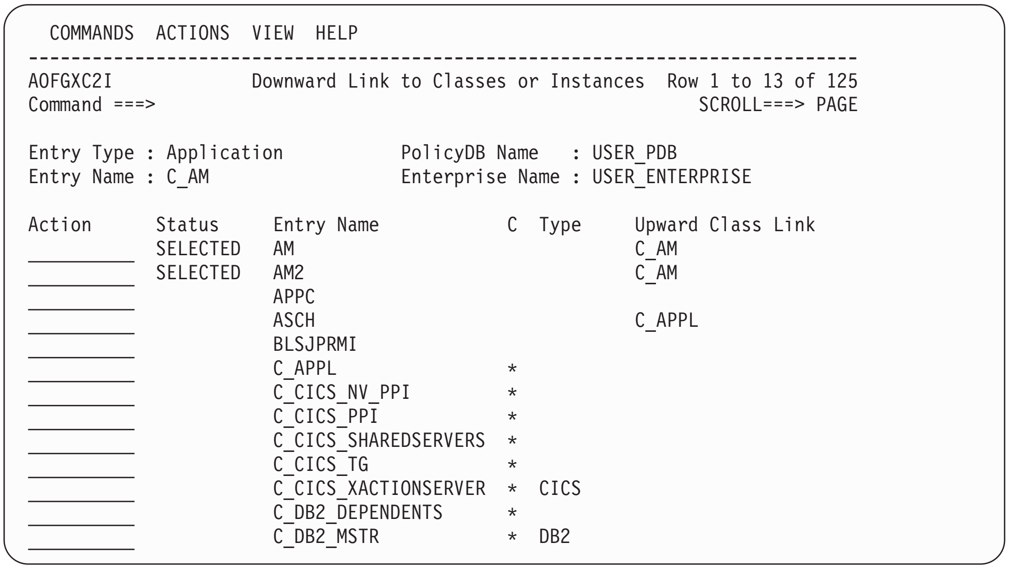
- Use the **Action** field to manage the link by entering:
  - Add (A) or Select (S) to link the class or instance to this class
  - Remove (M) to unlink the class or instance from this class
- The **Status** column indicates whether this class or instance has been linked up to the class in the list.
- The **Type** column indicates what application type the class in the list was defined as.
- The **Upward Class Link** column indicates the class, if any, that the entry in the list is currently upwardly linked to.

To avoid circular links, a check is made to ensure that the class you select is not already linked in the downward tree of the current class.

Refer to “Nested Classes” on page 29 for more details about nested class support.

DOWNWARD CLASS/INST Policy Item

If you select the DOWNWARD CLASS/INST policy item from the Policy Selection panel for Applications, the Downward Link to Classes or Instances panel is displayed, as shown in Figure 110 on page 167.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC2I             Downward Link to Classes or Instances  Row 1 to 13 of 125
Command ===>                                                 SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : C_AM                  Enterprise Name : USER_ENTERPRISE

Action      Status     Entry Name           C  Type     Upward Class Link
__________  SELECTED   AM                                C_AM
__________  SELECTED   AM2                               C_AM
__________             APPC
__________             ASCH                              C_APPL
__________             BLSJPRMI
__________             C_APPL               *
__________             C_CICS_NV_PPI        *
__________             C_CICS_PPI           *
__________             C_CICS_SHAREDSERVERS *
__________             C_CICS_TG            *
__________             C_CICS_XACTIONSERVER *  CICS
__________             C_DB2_DEPENDENTS     *
__________             C_DB2_MSTR           *  DB2
```

Figure 110. Downward Link to Classes or Instances Panel

This panel allows you to manage the downward links from an application class to classes or instances of the same type. These links allow the instances or classes to inherit any policy data that has been defined in this class or other classes further up in the class chain.

- Use the **Action** field to manage the links by entering:
  - Add (A) or Select (S) to link the class or instance to this class
  - Remove (M) to unlink the class or instance from this class
- The **Status** column indicates whether the entry in the list has been linked down to by this class.
- The **C** column indicates with an asterisk (*) that the entry is a class because this may not be obvious from the entry name.
- The **Type** column indicates what application type the entry was defined as. Note that an application class that you are working with can only be linked to entries of the same type as itself.
- The **Upward Class Link** column indicates the class, if any, that the entry in the list is currently upwardly linked to. If you select such an entry, you would change its class chain, so be careful not to break its data inheritance

To avoid circular links, a check is made to ensure that the class you select is not already linked in the upward tree of the current class.

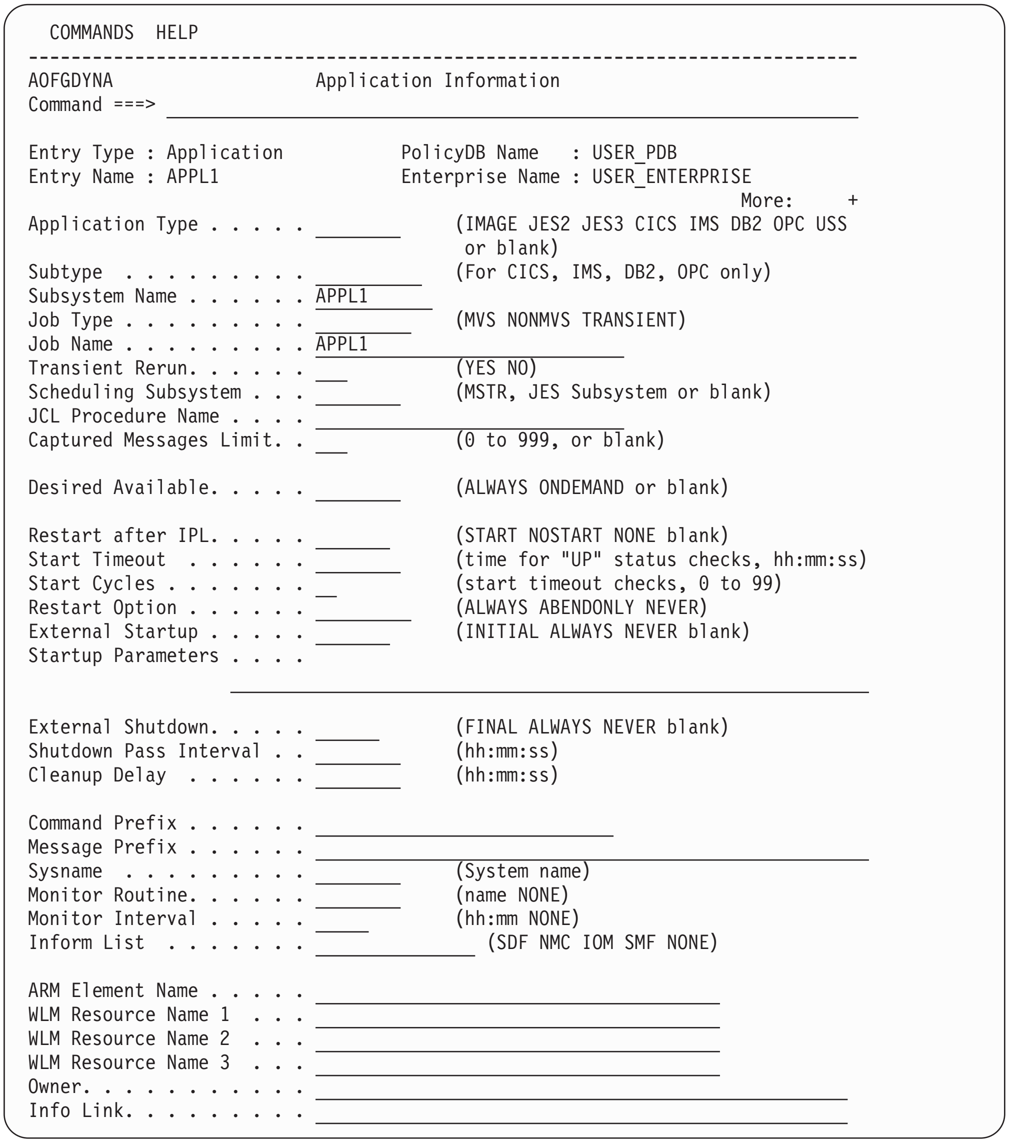
Both link panels show the application type because class-instance links are only allowed if there is no type mismatch in the class hierarchy, that is, either the specified types must be the same or no type is specified.

To avoid type mismatches, the types of the link partner or the inherited types are checked. A link is invalid if the resulting class chain includes type specifications that do not match (for example, a class or instance of type IMS and another class or instance of type CICS). Thus, if both link partners (the current entry and the linked class or instance) have a type specified it is checked that they are identical. If either link partner has no type specified, the class hierarchy that leads to this entry is scanned to check whether an inherited type may cause a mismatch. If this is the case, a detailed link error message that includes the names of the conflicting entries is displayed.

Refer to “Nested Classes” on page 29 for more details about nested class support.

APPLICATION INFO Policy Item

If you select the APPLICATION INFO policy item from the Policy Selection panel for Applications, the Application Information panel is displayed, as shown in Figure 111.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA                   Application Information
Command ===> _________________________________________________________________

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : APPL1                Enterprise Name : USER_ENTERPRISE
                                                               More:     +
Application Type . . . . . ________     (IMAGE JES2 JES3 CICS IMS DB2 OPC USS
                                         or blank)
Subtype  . . . . . . . . . __________   (For CICS, IMS, DB2, OPC only)
Subsystem Name . . . . . . APPL1_____
Job Type . . . . . . . . . _________    (MVS NONMVS TRANSIENT)
Job Name . . . . . . . . . APPL1____________________
Transient Rerun. . . . . . ___          (YES NO)
Scheduling Subsystem . . . ________     (MSTR, JES Subsystem or blank)
JCL Procedure Name . . . . _____________________________
Captured Messages Limit. . ___          (0 to 999, or blank)

Desired Available. . . . . ________     (ALWAYS ONDEMAND or blank)

Restart after IPL. . . . . _______      (START NOSTART NONE blank)
Start Timeout  . . . . . . ________     (time for "UP" status checks, hh:mm:ss)
Start Cycles . . . . . . . __           (start timeout checks, 0 to 99)
Restart Option . . . . . . _________    (ALWAYS ABENDONLY NEVER)
External Startup . . . . . _______      (INITIAL ALWAYS NEVER blank)
Startup Parameters . . . .
                  ____________________________________________________________

External Shutdown. . . . . ______       (FINAL ALWAYS NEVER blank)
Shutdown Pass Interval . . ________     (hh:mm:ss)
Cleanup Delay  . . . . . . ________     (hh:mm:ss)

Command Prefix . . . . . . ____________________________
Message Prefix . . . . . . ____________________________________________________
Sysname  . . . . . . . . . ________     (System name)
Monitor Routine. . . . . . ________     (name NONE)
Monitor Interval . . . . . _____        (hh:mm NONE)
Inform List  . . . . . . . _______________ (SDF NMC IOM SMF NONE)

ARM Element Name . . . . . ______________________________________
WLM Resource Name 1  . . . ______________________________________
WLM Resource Name 2  . . . ______________________________________
WLM Resource Name 3  . . . ______________________________________
Owner. . . . . . . . . . . __________________________________________________
Info Link. . . . . . . . . __________________________________________________
```

Figure 111. Application Information Panel

Any policy values that have been inherited from linked classes are displayed in a different color. You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line, moving the cursor to the field of interest and pressing Enter. You can override any inherited values by overtyping them. If you override any values for a class, they are individually applied to the current application and change color to turquoise to indicate this.

This panel displays the following information:

Application Type

Refer to “Application Type” on page 159 for more details.

Subtype

Refer to “Subtype” on page 159 for more details.

Subsystem Name

This gives a unique, eleven-character name to the application that you are defining. This name is used by SA z/OS automation. You can have multiple application instances with the same subsystem name, however, only one of them can be linked to a system. The subsystem name of an application class must be unique across the whole policy database, that is, it cannot be the same as the subsystem name of any other application instance or class.

You can change the subsystem name by overwriting the existing name. If the application instance is linked, the name is checked to be unique across all subsystem names in the policy database. If you want to change the subsystem name to a name that already exists, you first need to unlink the application instance from all application groups.

If no subsystem name is specified, the default is the value of the Entry name (if this name is not longer than 8 characters for applications of type IMAGE, and not longer than 11 characters for all other application types).

For applications of type IMAGE the subsystem name can only be an 8-character name.

> **Note:**
> Certain values are reserved for SA z/OS and cannot be used. Refer to the online help for a complete list.

Job Type

Refer to “Job Type” on page 159 for more details.

Job Name

Refer to “Job Name” on page 160 for more details.

Transient Rerun

Refer to “Transient Rerun” on page 160 for more details.

Scheduling Subsystem

Refer to “Scheduling Subsystem” on page 161 for more details.

JCL Procedure Name

Refer to “JCL Procedure Name” on page 161 for more details.

Captured Messages Limit

Specifies the maximum number of *captured messages* that will be saved for the SA z/OS command DISPINFO.

If the field is left blank the value can be inherited from a class, the application defaults (ADF), or the system’s default definition (SDF). If neither of these are specified a default value of 0 is used.

Desired Available

Specifies how the desired status of the resource is determined in the absence of any propagated desired status votes on the resource.

You can specify the following values:

ALWAYS
The resource's desired status is set to Available and it is started once its dependencies are satisfied. This is the same behavior as in earlier releases of SA z/OS.

ONDEMAND
The resource's desired status is set to Unavailable and it is not started. For SA z/OS to start the resource it must be propagated a MakeAvailable vote. This could come from an operator request, a service period, or a Move or Server group.

Note: A dependency from a Desired Available: ALWAYS resource is not sufficient to cause a Desired Available: ONDEMAND resource to be started.

blank If the field is left blank the value can be inherited from a class or the system's default definition (SDF). If no value can be found, ALWAYS is used.

Restart after IPL
Specifies how SA z/OS determines the initial status of the application on the first occasion that SA z/OS is started after a z/OS IPL or an automation status file reallocation.

Reallocating the automation status file has the same effect as a z/OS IPL because SA z/OS stores the last IPL date and time in the automation status file. When the automation status file is deleted, SA z/OS assumes that the next startup is the first startup following a z/OS IPL.

You can specify four values for Restart after IPL:

START
The application is always startable when the start dependency is fulfilled. The application status is set to DOWN during initialization. It is recommended that you specify YES for critical applications such as JES, VTAM, and TSO. Replying NOSTART to message AOF603D (specify initialization options) overrides this setting.

NOSTART
This enforces a STOP request with priority FORCE to be injected during SA z/OS initialization. This results in a desired status of UNAVAILABLE and an observed status of SOFTDOWN, which produces a compound status of SATISFACTORY. If the observed status before IPL was HARDDOWN this status remains in HARDDOWN, which produces a compound status of PROBLEM.

blank This indicates that the application is startable, depending on its status. If the status is STOPPED, BROKEN or CTLDOWN, it is not changed and the application is not started. Any transient status (for example, BREAKING) is changed to the corresponding final status (for example, BROKEN). Any other status is changed to DOWN during initialization and, if the INITSTART flag is set to YES, the application is started when all its relationships are fulfilled.

NONE
This value is the same as blank except that this application, if it is an instance, does not inherit the value from its related application class.

Start Timeout
The start timeout interval that begins when SA z/OS issues a start command for an application. If the application fails to reach UP status in this time, an application monitor check is performed and the application's status is set to either STARTED2 if the application has been started or to INACTIVE if the application is not found. Message AOF571 is issued with a description of the situation. The default value for Start Timeout is 2 minutes.

Start Cycles
Specifies the number of times to cycle the start timeout period before posting the subsystem as a problem. The default value for Start Cycles is 1.

Restart Option
The circumstances that the application should be restarted under. The options are:

Always
The application is restarted if it was terminated without using the INGREQ command. Note that, if the advanced automation option AOFRESTARTALWAYS is set to 0, the application restart thresholds are checked. The application is restarted only if it has not exceeded the critical threshold. By default the application is restarted every time it is stopped outside the control of SA z/OS.

Abendonly
The application is eligible for restart only if it abnormally ends. The application is restarted only if its critical error threshold has not been exceeded.

Never The application is never eligible for restart by SA z/OS.

You can use this option to force a specific application state. For example, if an application is defined with the restart option set to Always and an operator tries to shut down the application without using the INGREQ command, the application will be restarted when it has completed its shutdown. The AOFRESTARTALWAYS advanced automation option can be used to modify this behavior.

External Startup
Specifies whether the application is started externally or via a specified startup procedure.

INITIAL
The application will only initially be started externally.

ALWAYS
The application will always be started externally.

NEVER
The application will never be started externally.

blank Indicates that the value can be inherited from an application CLASS entry.

For applications of type IMAGE where the monitor routine is INGMTSYS the value is forced to ALWAYS.

Startup Parameters
Specifies the desired subsystem startup parameters. These values are added onto the MVS START command for the subsystem. For example, the desired start command for VTAM may be:

```
S VTAM,,,(LIST=00)
```

Therefore, if this command is to be submitted from SA z/OS the parameters field should contain:

```
,,,(LIST=00)
```

Whatever is entered in the parameters field is appended directly to the MVS START command for the subsystem. Therefore, the commas preceding `(LIST=00)` are necessary. You may include system symbols and system automation symbols. You may also include runtime variables , like SUBSAPPL, SUBSJOB, SUBSPROC, SUBSSHUTDLY and others.

External Shutdown
Specifies whether the application is stopped externally or via a specified shutdown procedure.

FINAL
The application will be stopped externally if the external supporting resource (parent) is also being stopped.

ALWAYS
The application will always be stopped externally.

NEVER
The application will never be stopped externally.

blank Indicates that the value can be inherited from an application CLASS entry

For applications of type IMAGE where the monitor routine is INGMTSYS, the value is forced to ALWAYS.

Shutdown Pass Interval
The time delay between attempts to perform automated shutdown of an application. Automation can perform multiple passes of shutdown attempts. This value relates to shutdown commands specified for the application. The MDFYSHUT command can be issued on shutdown passes to change the timing between consecutive passes.

Cleanup Delay
The delay between application monitoring checks performed by SA z/OS after the final termination message (TERMMSG FINAL=YES) for an application is processed. The termination procedure is as follows:

1. When SA z/OS receives the final termination message for an application, it waits for the period defined in the AOFPAUSE global variable before performing its first application monitoring check. (See *IBM Tivoli System Automation for z/OS Customizing and Programming* for more information on AOFPAUSE.)
2. If the application monitoring check shows that the application is still active, it will wait for the period specified in Cleanup Delay and try again. This will be repeated if necessary.
3. If, after two checks, an application monitor check still finds the application to be active, the application will be posted to ZOMBIE status and periodically rechecked at increasing intervals.

   The default value for the Cleanup Delay is 12 seconds.

Command Prefix
This is the console prefix character for the application. Many applications, such as JES2, JES3, and NetView support the use of a command prefix

character which identifies a command as belonging to that application. If such support is available for this application, enter the corresponding command character.

Any command that would normally be issued at a z/OS console should be prefixed with MVS. NetView commands to control NetView automation do not require the MVS prefix.

You may include system symbols and system automation symbols, see "Assigning System Automation Symbols (AOCCLONE)" on page 27.

> **Note:**
> The single quotation character (′) is reserved for SA z/OS and should not be used.

Message Prefix
Specifies one or more message prefixes for this application. You only have to specify these message prefixes if the application issues messages that do not contain the MVS job name. These message prefixes must be unique for this application.

Sysname
The system name for the subsystem. This is only required for JES subsystems where the Sysname for the subsystem is not the same as the MVS Sysname. If Sysname is not specified, it will default to the name specified in the SYSTEM INFO policy item of the System policy object.

Monitor Routine
The name of the application monitor routine that is used to monitor the application's status.

The following specifications are valid:

AOFADMON
Monitors the status via MVS D A commands. This should be avoided.

AOFAPMON
Routine used to monitor a PPI receiver.

AOFATMON
Monitors the status of a task in the NetView environment.

AOFCPSM
Dedicated routine used to monitor SA z/OS Processor Operations.

AOFNCMON
Routine used to monitor the connection between the NMC focal point NetView and the NMC server application.

AOFUXMON
Routine used to monitor USS applications.

INGMTSYS
Dedicated routine used to monitor IMAGE applications for BCPII usage.

ISQMTSYS
Dedicated routine used to monitor SA z/OS processor operations target system resources.

INGPJMON
Monitors the status via ASCB checking.

INGPSMON
Monitors the status via IEFSSI service.

INGVMON
Monitors the status of a VOST via LIST STATUS=VOST.

NONE
No monitoring routine is to be used.

username
A monitoring routine that has been defined and created by the user.

> **Note:**
> If a user-defined routine is not found by automation, the SA z/OS routine INGPJMON is used except for applications of type IMAGE, when INGMTSYS is the default.
>
> For non-MVS applications, make sure that either a user-defined monitoring routine or NONE is specified.

If this application is linked to an application CLASS entry that has the Monitor Routine specified, the CLASS policy takes precedence over the Environment Setup policy.

Monitor Interval
This is the amount of time between application subsystem monitoring cycles. Monitoring may be suspended by specifying NONE.

Inform List
This field allows you to specify:

- Where the application is registered to
- Where all status changes are propagated to
- Which SMF records are written

Valid receivers are SDF, NMC, IOM, and SMF. You can specify more than one receiver. The list of values can be blank or comma separated.

Neither system automation symbols (AOCCLONEs) nor system symbols are allowed.

If the field is left blank the value is obtained from the systems default definition (see “System Defaults Policy Object” on page 288).

If the resource should not be registered at all, specify NONE. This prevents inheritance from the system's default definitions (SDF).

ARM Element Name
An application defined to MVS Automatic Restart Manager must have an element name that is unique within each sysplex where the application runs. Each element name consists of up to 16 alphanumeric and national characters including @,#,$ and _ (underscore) after clone resolution, provided the first character is not numeric. Clone resolution occurs at automation control file load.

You may include system symbols and system automation symbols. For more details, see “Assigning System Automation Symbols (AOCCLONE)” on page 27.

The element name can be:

Constant

The element name is the same, regardless of which system the application is running on. This is required for applications that are restartable on multiple systems in the sysplex. An application with a constant Automatic Restart Manager element name may at the same time have a cloned job name.

> **Note:**
> An application with a cloned job name and a constant Automatic Restart Manager element name can be started on only one system within a sysplex.

For applications of type IMAGE the value is forced to blank.

Cloned

The application has a different element name on each system. The element name is derived from some attribute that is unique to the system that the application runs on. Applications with cloned element names must be defined to MVS Automatic Restart Manager in such a way that MVS will only attempt to restart them on the same system and not move them to other systems in the sysplex.

Applications with cloned element names may have constant job names. This can be useful if you want to have the same application running on multiple systems with Automatic Restart Manager recovery enabled.

For example, you have a USERPROC that runs on all your systems and you want Automatic Restart Manager recovery for it. Automatic restart management requires that USERPROC have a different element name on each system. Defining USERPROC to SA z/OS with a constant job name, but with a cloned element name (and perhaps some parameters to define the element name) allows you to define a single SA z/OS application with a different element name on each system.

WLM Resource Name 1–3

These are the names of the WLM Resources that are associated with the application.

The *same* WLM resource name can be used for *multiple* applications. If any of these applications are in the UP or ENDED automation status, the WLM resource status is set to ON. When all the applications that are associated with the same WLM resource name are in any other status, the resource is set to OFF.

You may include system symbols and system automation symbols, see “Assigning System Automation Symbols (AOCCLONE)” on page 27.

You can specify up to three different WLM resource names, but only one per field.

No WLM resource names are accepted for applications of type IMAGE.

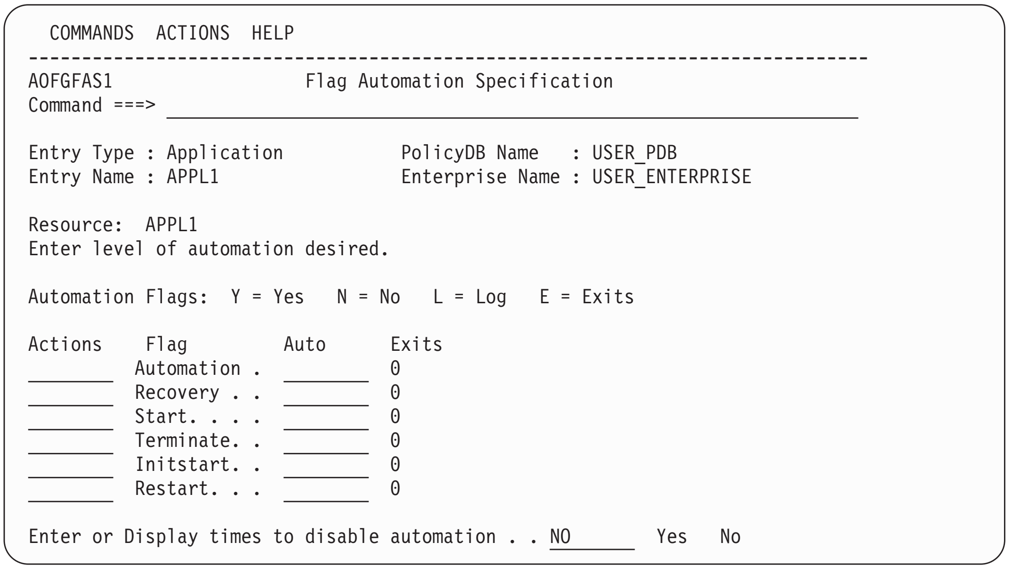
All three WLM resource names are inherited together. Thus if you specify one for an instance and another for a class, this is not inherited.

Owner
This specifies information for the operator about who to contact in case of error.

Info Link
This field can be used to specify a location (for example, a URL) where additional information about the application can be found.

AUTOMATION FLAGS Policy Item

The Flag Automation Specification panel (Figure 112) is displayed if you select policy item AUTOMATION FLAGS from the Policy Selection panel for applications. For more details about automation flags, refer to “Automation Flags” on page 20.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGFAS1                 Flag Automation Specification
Command ===> _________________________________________________________________

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

Resource:  APPL1
Enter level of automation desired.

Automation Flags:  Y = Yes   N = No   L = Log   E = Exits

Actions    Flag        Auto      Exits
________  Automation .  ________  0
________  Recovery . .  ________  0
________  Start. . . .  ________  0
________  Terminate. .  ________  0
________  Initstart. .  ________  0
________  Restart. . .  ________  0

Enter or Display times to disable automation . . NO______  Yes   No
```

Figure 112. Flag Automation Specification Panel

This panel allows you to specify:

- The type of automation mode
- Whether automated operator functions will operate in log assist mode
- Whether user-provided exit routines are to be called to determine whether automation is to continue
- When automation occurs

There are six types of automation flags supported by SA z/OS:

Automation
Is the global automation flag that must be YES for the remaining flags to be used. If it is NO, there is no automation for the resource.

Recovery
Indicates whether recovery of the resource (actions on messages that are neither for the initialization nor the ending of the resource) is allowed.

Start Indicates whether the start command and actions or responses to initialization for the resource are allowed.

Terminate
Indicates whether the shutdown command and actions on or responses to ending messages are allowed.

InitStart
Indicates whether a resource can be started when its state is DOWN (that is, it controls initial starting of the resource).

Restart
Indicates whether the restarting of a resource that abnormally ended or was automatically shut down is allowed.

For each flag you may specify:

Actions
Enter S in the **Actions** field to select the flag to define or delete user-defined exit routines.

Auto Controls whether particular automated responses are allowed:

YES Indicates that automation is active.

NO Indicates that automation is not active.

LOG Indicates that LOG assist mode is active, which means that actions that should be taken under the chosen type of automation do not occur, but messages describing what would have happened are written to the NetView log.

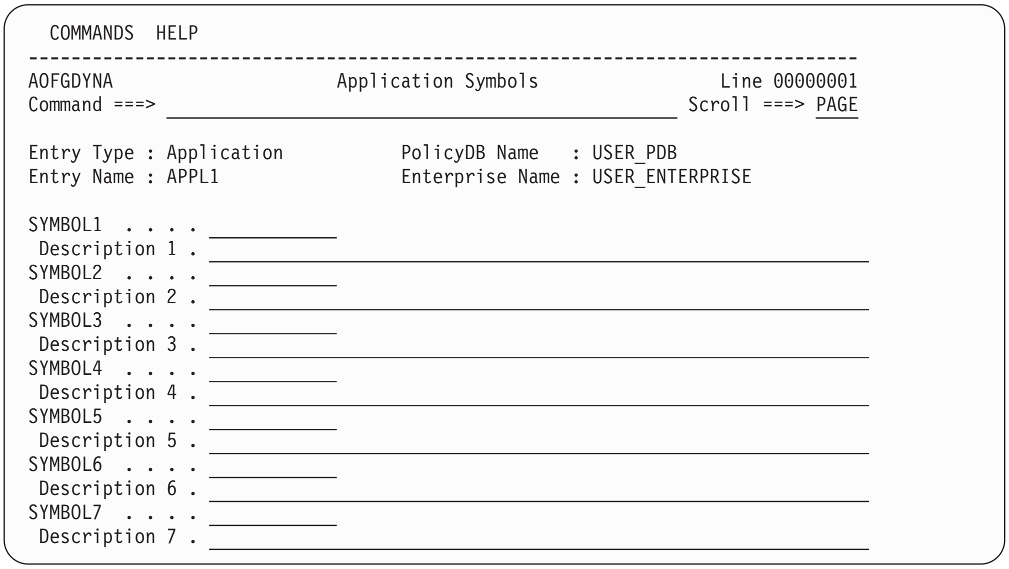
EXITS Indicates that there are one or more user-provided exit routines defined.

The **Exits** field displays the number of exits defined.

If you want to enable automation but specify times when it should be disabled, you can enter YES in the final input field. Another panel will be displayed where you can specify one or more time ranges when automation will be disabled.

APPLICATION SYMBOLS Policy Item

If you select the APPLICATION SYMBOLS policy item from the Policy Selection panel for Applications, the Application Symbols panel is displayed, as shown in Figure 113 on page 178.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA                     Application Symbols                 Line 00000001
Command ===> ________________________________________________ Scroll ===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

SYMBOL1  . . . . ____________
 Description 1 . ______________________________________________________________
SYMBOL2  . . . . ____________
 Description 2 . ______________________________________________________________
SYMBOL3  . . . . ____________
 Description 3 . ______________________________________________________________
SYMBOL4  . . . . ____________
 Description 4 . ______________________________________________________________
SYMBOL5  . . . . ____________
 Description 5 . ______________________________________________________________
SYMBOL6  . . . . ____________
 Description 6 . ______________________________________________________________
SYMBOL7  . . . . ____________
 Description 7 . ______________________________________________________________
```

Figure 113. Application Symbols Panel

This panel allows you to enter or update application symbols that can be set to various application-specific values. These can then be substituted using the runtime variables SUBSSYMBOL1 through SUBSSYMBOL9. Those variables can be used in a similar fashion to runtime variables like SUBSSPARM, SUBSSCHEDSS, SUBSJOB, etc.

To specify that SUBSSYMBOL value `x` is to be substituted, enter `&SUBSSYMBOLx` at the appropriate point in the field. You must specify only the leading ampersand (&).

SYMBOL*x*

Allows you to specify the value of the application symbol. You can enter any character but you should avoid using any SA z/OS reserved runtime variables, such as &SUBSJOB or others.

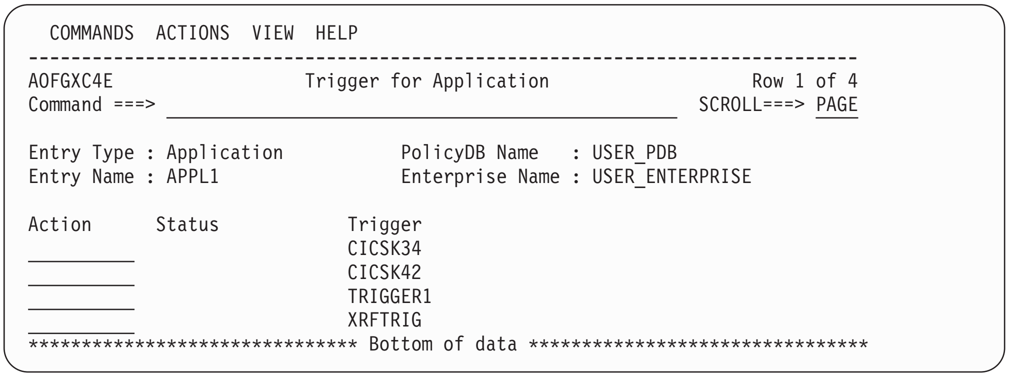
Note: The entered data is not automatically translated to upper case.

Description

Allows you to specify a description of the application symbol.

TRIGGER Policy Item

The Trigger for Application panel (Figure 114 on page 179) is displayed if you select policy item TRIGGER from the Policy Selection panel for applications. For more details about triggers, refer to “Event and Trigger Support” on page 25.



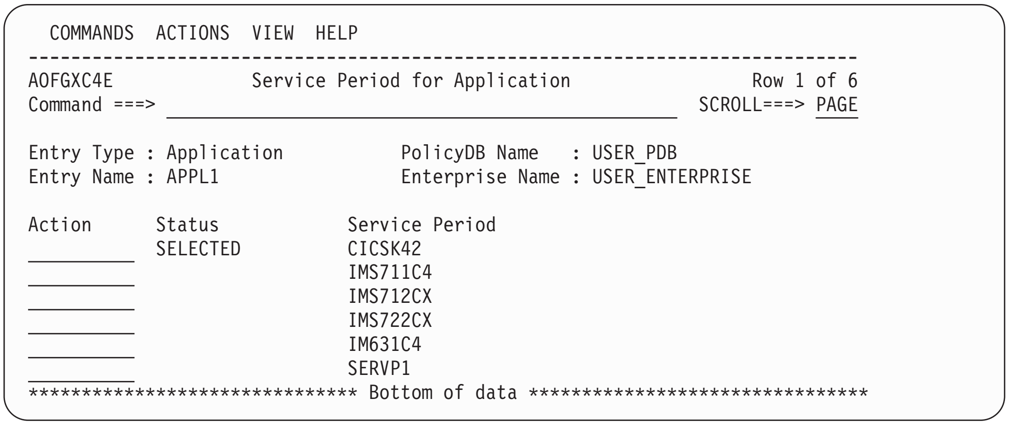
```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E                Trigger for Application                    Row 1 of 4
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : APPL1                Enterprise Name : USER_ENTERPRISE

Action      Status           Trigger
__________                   CICSK34
__________                   CICSK42
__________                   TRIGGER1
__________                   XRFTRIG
****************************** Bottom of data ********************************
```

Figure 114. Trigger for Application Panel

Here you can add triggers to or remove them from an application.

SERVICE PERIOD Policy Item

If you select an application's SERVICE PERIOD policy item, you reach the Service Period for Application panel. Figure 115 shows an example.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E           Service Period for Application                  Row 1 of 6
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : APPL1                Enterprise Name : USER_ENTERPRISE

Action      Status           Service Period
__________  SELECTED         CICSK42
__________                   IMS711C4
__________                   IMS712CX
__________                   IMS722CX
__________                   IM631C4
__________                   SERVP1
****************************** Bottom of data ********************************
```

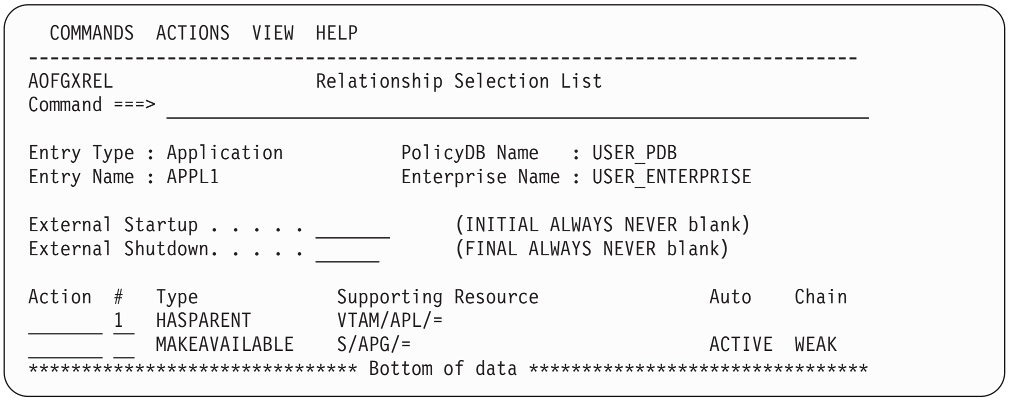
Figure 115. Service Period for Application Panel

Here you can add service periods to or remove them from an application.

RELATIONSHIPS Policy Item

The Relationship Selection List panel (Figure 101 on page 143) is displayed if you select policy item RELATIONSHIPS from the Policy Selection panel for applications. It shows information about existing relationships and allows you to define new relationships.

The ACTION pull-down choice offers actions to edit and delete selected relationships. The **D**elete command immediately deletes the relationship. The **E**dit command leads to the Define Relationship panel (as shown in Figure 116 on page 180).



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXREL                   Relationship Selection List
Command ===> ____________________________________________________________________

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : APPL1                Enterprise Name : USER_ENTERPRISE

External Startup . . . . . _______      (INITIAL ALWAYS NEVER blank)
External Shutdown. . . . . ______       (FINAL ALWAYS NEVER blank)

Action  #   Type            Supporting Resource              Auto    Chain
_______ 1_  HASPARENT       VTAM/APL/=
_______ __  MAKEAVAILABLE   S/APG/=                          ACTIVE  WEAK
******************************* Bottom of data ********************************
```

Figure 116. Relationship Selection List Panel

This panel displays information about existing relationships. In all of the relationships for the current resource, this resource is the dependent resource. The columns have the same meaning as described in “Defining Relationships” on page 143.

If the application is linked to a class, the **External Startup** and **External Shutdown** fields in the subheader area show inherited data individually, however the relationships are inherited only as a whole if no relationships are defined for the child object.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line, moving the cursor to the field of interest and pressing Enter.

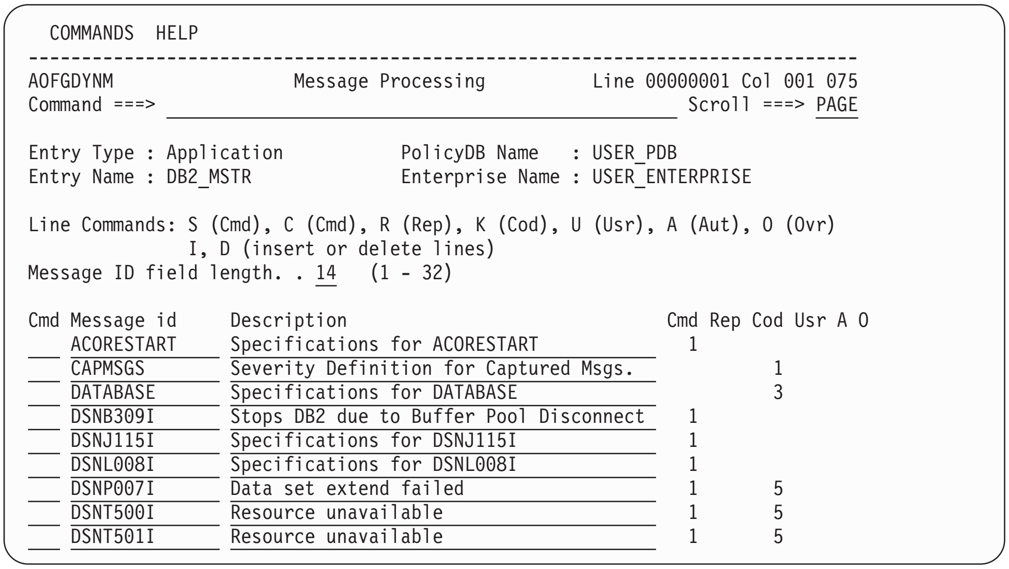
If the application has inherited relationships from a class, only these are displayed in a different color. If you change one of these relationships, all the inherited relationships become specific to that particular class or instance (and thus change color to reflect this).

Entering the NEW command from this panel to define new relationships takes you to the Define Relationship panel described in “Defining Relationships” on page 143.

The External Startup and External Shutdown fields are the ones from “APPLICATION INFO Policy Item” on page 168 (see “Startup Parameters” on page 171 and “External Shutdown” on page 172 for details).

MESSAGES/USER DATA Policy Item

If you select policy item MESSAGES/USER DATA from the Policy Selection panel for Applications (Figure 108 on page 162), the Message Processing panel shown in Figure 117 on page 181 is displayed.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNM                 Message Processing         Line 00000001 Col 001 075
Command ===>                                                 Scroll ===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : DB2_MSTR              Enterprise Name : USER_ENTERPRISE

Line Commands: S (Cmd), C (Cmd), R (Rep), K (Cod), U (Usr), A (Aut), O (Ovr)
               I, D (insert or delete lines)
Message ID field length. . 14   (1 - 32)

Cmd Message id     Description                              Cmd Rep Cod Usr A O
___ ACORESTART     Specifications for ACORESTART              1
___ CAPMSGS        Severity Definition for Captured Msgs.             1
___ DATABASE       Specifications for DATABASE                        3
___ DSNB309I       Stops DB2 due to Buffer Pool Disconnect    1
___ DSNJ115I       Specifications for DSNJ115I                1
___ DSNL008I       Specifications for DSNL008I                1
___ DSNP007I       Data set extend failed                     1       5
___ DSNT500I       Resource unavailable                       1       5
___ DSNT501I       Resource unavailable                       1       5
```

Figure 117. Message Processing Panel

On the Message Processing panel, you specify how SA z/OS should react if the application issues the specified message. If the application is linked to a class, the panel shows both instance and class data sorted alphabetically, with the inherited data shown in a different color. For more details see “Nested Class Details” on page 190. You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line, moving the cursor to the field of interest and pressing Enter.

On this panel you can enter the FIND (abbreviated to F) on the command line to search the displayed data for a specific string, for example, `f abc`. All occurrences of the string are highlighted and the cursor is positioned at the start of the first one. You can use the RFIND function key (PF5) to move to the next occurrence of the string.

Use the RESET command to remove the find highlighting. If you enter the RFIND command at the command line it repeats the last find.

Use the CANCEL command (abbreviated to C) to exit the panel without saving any changes to the data.

You can define message responses for each message ID. By default, the first 14 characters of the message ID is shown. The length of the description that is displayed depends on your screen width (for a 24x80 screen, it is 40 characters).

You can change the length of the message ID field with the **Message ID field length** field. Note that this value is not saved across sessions. If you increase the message ID field length, you can view any information on the right of the panel that is no longer displayed by scrolling right with the PF11 key.

You can enter different message responses in the **Cmd** column. Depending on what you type here, SA z/OS displays a follow-on panel where you further specify SA z/OS’s automated reaction to the message. The message ID must not end with an asterisk and cannot have any commas in it.

For each of the automated responses, an entry in the NetView automation table is required that specifies which routine SA z/OS should be used to process your specified action. SA z/OS is able automatically to build the NetView AT entries for all CMD, REPLY, AUT, and OVR actions as well as for certain USR and COD actions.

Information about how to manage the NetView automation table is given in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

The following actions are available:

CMD To issue commands as a response to a message, see “CMD Processing” on page 183.

REP To issue replies, see “Reply Processing” on page 185.

COD If you want check for certain codes within the message. If the codes are contained in the message, you can enter a value that is returned to the calling CLIST, see “Code Processing” on page 186.

USR You can use the associated panel to let SA z/OS perform user-specified actions if the current message requires more specific automation than provided by the generic routines (see “User Defined Data” on page 188).

Using the ACFFQRY common routine from within a CLIST, you can retrieve the keywords entered into this panel and process them as needed. Syntax rules for specifying user data are available in the online help. For information on how to use ACFFQRY in a CLIST refer to ACFFQRY in *IBM Tivoli System Automation for z/OS Programmer's Reference*.

AUT You can use the associated panel to specify messages that indicate the status of a resource, see “AUTO Processing” on page 188. Once messages have been defined, they are automatically added to the NetView automation table.

OVR This allows you to override a default NetView automation table entry generated for a message, see “Overriding Message Automation” on page 189.

Notes:

1. AUT cannot be defined in combination with OVR.
2. The AUT action allows you to define Ignore or Suppress, which will prevent the building of an AT entry for the message.

Reserved Message IDs

With the following message IDs and also with any of the SA z/OS defined keywords you can invoke a special SA z/OS automation:

- ACORESTART (which is associated with action *CMD*)
- AMRFSHORT
- AMRFFULL
- AMRFCLEAR
- CAPMSGS
- DN_*susbsystem_name*
- INGALERT
- LOGREC
- MVSDUMPTAKEN
- MVSDUMPRESET

- MVSDUMP
- SMFDUMP
- SYSLOG
- UP_*susbsystem_name*
- VTAMUP
- WTORS
- Any of the system operations statuses

You can also specify a short description of the action that the response performs.

Note:
For product automation (that is, DB2, CICS, IMS, TWS) there are further keywords defined. For details see the relevant manual:
- *IBM Tivoli System Automation for z/OS CICS Automation Programmer's Reference and Operator's Guide*
- *IBM Tivoli System Automation for z/OS IMS Automation Programmer's Reference and Operator's Guide*
- *IBM Tivoli System Automation for z/OS TWS Automation Programmer's Reference and Operator's Guide*

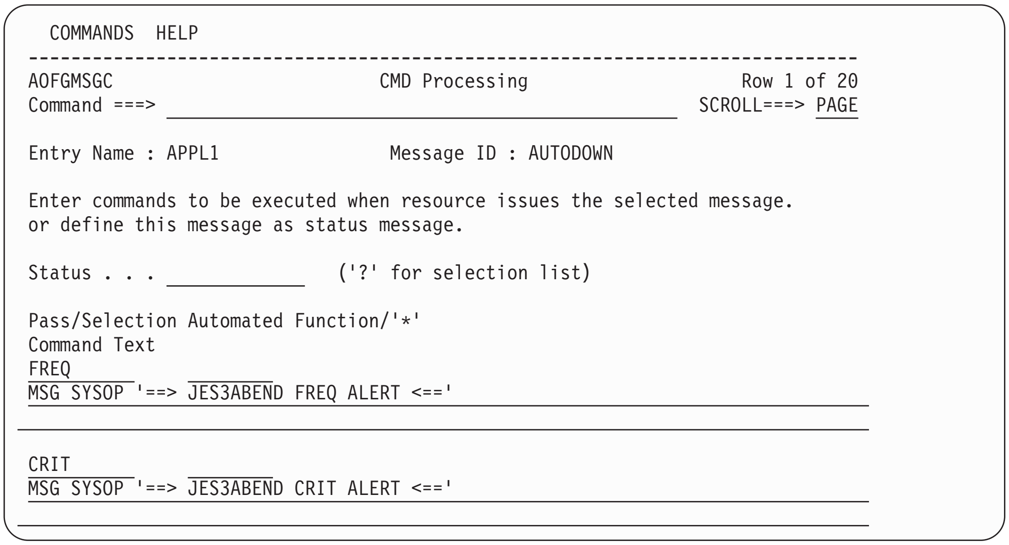
Product automation has the following reserved message IDs"
- For CICS:
  - ABCODES
  - ABCODESYSTM
  - ABCODETRAN.*xxxxx* (where *xxxxx* is the optional transaction name).
  - CICSINFO
  - HEALTHCHK
  - LISTSHUT
  - RCVRTRAN.*xxxxx* (where *xxxxx* is the optional transaction name).
- For IMS:
  - ABCODETRAN.*xxxxx* (where *xxxxx* is the optional transaction name).
  - ABCODEPROG.*xxxxx* (where *xxxxx* is the program name).
  - IMSINFO
  - OLDS
  - RESTARTABORT
  - TCO
  - TCOMEMBERS

For a list of keywords that no AT entries are built for, see "Restricted Message IDs" in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

For more information on how to use these various methods of message processing refer to *IBM Tivoli System Automation for z/OS Customizing and Programming*.

CMD Processing

SA z/OS displays the CMD Processing panel (Figure 118 on page 184) if you entered the action code CMD in the **Action** column on the Message Processing panel.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGC                         CMD Processing                   Row 1 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Name : APPL1                Message ID : AUTODOWN

Enter commands to be executed when resource issues the selected message.
or define this message as status message.

Status . . .              ('?' for selection list)

Pass/Selection Automated Function/'*'
Command Text
FREQ
MSG SYSOP '==> JES3ABEND FREQ ALERT <=='


CRIT
MSG SYSOP '==> JES3ABEND CRIT ALERT <=='

```

Figure 118. CMD Processing Panel

Use this panel to enter commands that are executed when the application issues the previously specified message. If you specify one of the SA z/OS provided automation commands (described in *IBM Tivoli System Automation for z/OS Operator's Commands*), they should be queued using the NetView CMD LOW command. This ensures that task global variables in use by the calling automation procedure are not changed by the calling command.

For a list of run time variables that can be substituted into each command prior to execution, see ACFCMD in *IBM Tivoli System Automation for z/OS Programmer's Reference*.

The **Status** field on the CMD Processing panel defines a resource status that is indicated by the Message ID. Valid values are displayed in a selection list if either a question mark or unknown status information is entered.

Any value contained in the selection list can be directly defined in the **Status** field. Alternatively, this field allows you to mark the message to be captured, ignored, or suppressed. If it is to be ignored or suppressed, a NetView Automation Table entry is not generated. No data is accepted in this field if an override (OVR) action exists.

The value entered in the **Pass/Selection** field defines when the command or reply should be issued, as follows:

Pass Enter a number from 1 through 99 or an asterisk (*) in this field to specify a pass value. During the automation control file build, this value is prefixed with PASS to create the selection criterion for the CMD or REP entry.

For more information relating to for Programmer's Reference.

Note: For more information about pass processing see the description of the PASSES parameter for the ISSUEACT generic routine in *IBM Tivoli System Automation for z/OS Programmer's Reference*. Pay attention when specifying a number because this is treated as a selection.

Selection

An expression other than a pass value is considered a selection value and is not altered during the build process. A selection value can be a blank or any name that does not begin with a number or contain imbedded blanks, commas, quotes, or parentheses.

Note: For commands issued by SA z/OS during startup or shutdown, an attempt is made to issue only the commands relevant to the particular start or stop type. SA z/OS looks for command definitions with a selection field that matches the start or stop type. Commands that are defined without a selection value are issued or selected for each start or stop type.

Automated Function/'*'

Enter the name of the automated function that the command is to run under. If you leave the field blank, the function defaults to the one currently running the command list. SA z/OS automated functions are mapped to NetView automation operators. This is documented in Table 7 on page 292.

Note:

Entering a '*' enables return code checking for the specified command. This means that processing continues if the command returns a return code of 0. In all other cases processing is stopped immediately. If running in LOG assist mode, return code checking has no effect because the commands are not executed.

Command Text

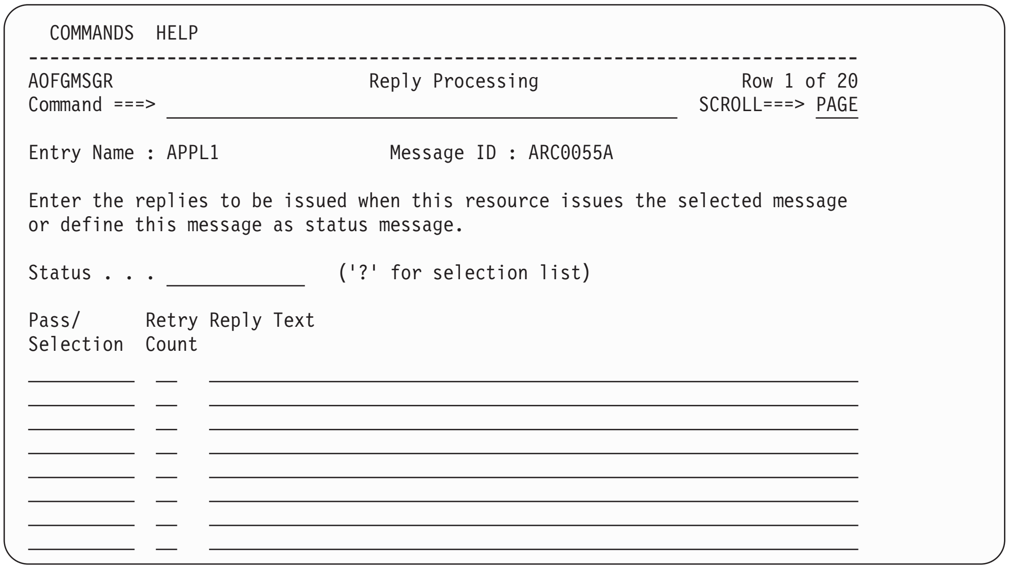
Enter the command or command list to be issued. A system command must be prefaced with MVS.

Pass or selection values can be repeated for several text entries. Each set of entries with the same pass or selection value forms a command group.

To process this message an entry in the NetView automation table is generated that uses the ISSUEACT generic routine. For more details, see ISSUEACT in *IBM Tivoli System Automation for z/OS Programmer's Reference*.

Reply Processing

SA z/OS displays the Reply Processing panel if you entered the action code REP in the **Action** column on the Message Processing panel.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGR                        Reply Processing                 Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Name : APPL1                Message ID : ARC0055A

Enter the replies to be issued when this resource issues the selected message
or define this message as status message.

Status . . . _____________    ('?' for selection list)

Pass/      Retry Reply Text
Selection  Count
__________  __   _____________________________________________________________
__________  __   _____________________________________________________________
__________  __   _____________________________________________________________
__________  __   _____________________________________________________________
__________  __   _____________________________________________________________
__________  __   _____________________________________________________________
__________  __   _____________________________________________________________
__________  __   _____________________________________________________________
```

Figure 119. Reply Processing Panel

Use this panel to enter the replies to be issued when this resource issues the previously specified message. For a list of run time variables that can be substituted into each reply prior to execution, see ACFREP in *IBM Tivoli System Automation for z/OS Programmer's Reference*.

The **Status** field on the Reply Processing panel is the same as described in “CMD Processing” on page 183.

The use of the **Pass/Selection** field is similar to that described in “CMD Processing” on page 183. Pass or selection values can be repeated for several text entries. Each set of entries with the same pass or selection forms a reply group.

To process this message an entry in the NetView automation table is generated that uses the ISSUEACT generic routine. For more details, see ISSUEACT in *IBM Tivoli System Automation for z/OS Programmer's Reference*.

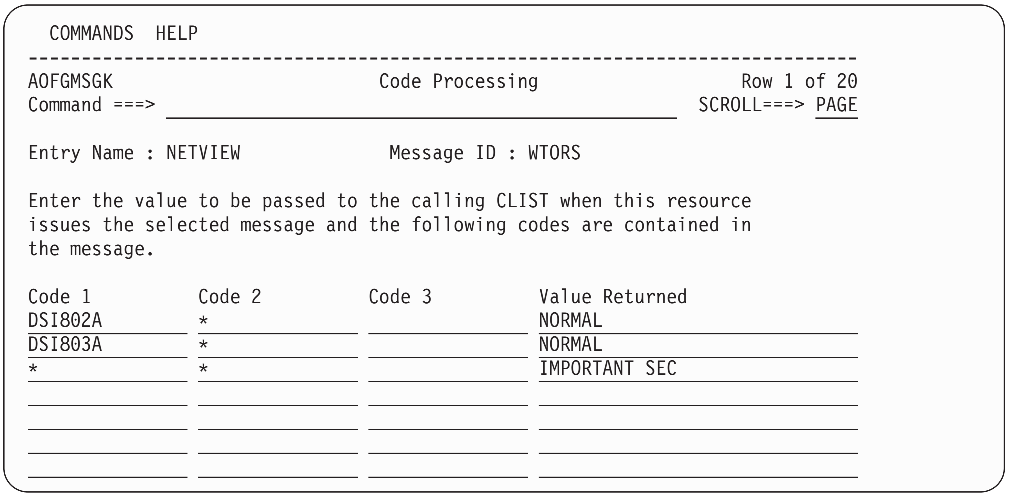
Additionally, there are the following fields:

Retry Count
Enter a number in the range from 1 through 99 in this field to define the number of retry attempts to be made before the common routine decides that a reply cannot be issued. A retry occurs when the outstanding reply number is not available. There is a two second time delay between each retry. The default retry count is 0.

Reply Text
Type the actual reply to the message in this field. Enter it in the same format as you would do from an MVS console (without the R and the reply number). To specify a null reply, type NULL in this field.

Code Processing

SA z/OS displays the panel shown in Figure 120 on page 187 if you enter the action code COD in the **Action** column on the Message Processing panel.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGK                         Code Processing                 Row 1 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Name : NETVIEW             Message ID : WTORS

Enter the value to be passed to the calling CLIST when this resource
issues the selected message and the following codes are contained in
the message.

Code 1          Code 2          Code 3          Value Returned
DSI802A         *                               NORMAL
DSI803A         *                               NORMAL
*               *                               IMPORTANT SEC
```

Figure 120. Code Processing Panel

Use this panel to enter the value to be passed to the calling CLIST when the resource issues the previously-specified message, and the codes you specify in this panel are contained in the message. Your CLIST can call the common routine CDEMATCH to compare data with the information in the CODE fields. You can use the override (OVR) action to define the associated AT entry.

For non-predefined message automation use the CDEMATCH routine in the NetView automation table to process this kind of message. For more details, see CDEMATCH in *IBM Tivoli System Automation for z/OS Programmer's Reference*.

The entry fields in the Code Processing panel have the following purpose:

Code 1, 2, 3
: Data entered in these fields is compared with data from the user command list (CLIST). The data in the command list is typically retrieved from the message. Therefore if the codes defined here match the codes from the message text, a value is returned to the command list. You need not use all three **Code** fields.

Value Returned
: If all the CODE fields that are defined match those from the command list, the value in this field is returned to the command list to be used for decision making.

Code definitions can be used to modify the processing of termination messages in the TERMMSG generic routine. If the code values that have been passed as parameters to TERMMSG match a code definition, the first token of the returned value is used as the status that the application is placed in by TERMMSG.

A second token in the returned value is used as the start type for the application's next startup. The next start type is set with the INGSET command.

Example: TERMMSG is called with CODE1=ABENDED and CODE2=S222 by the termination message of an application that has the following codes defined for it:

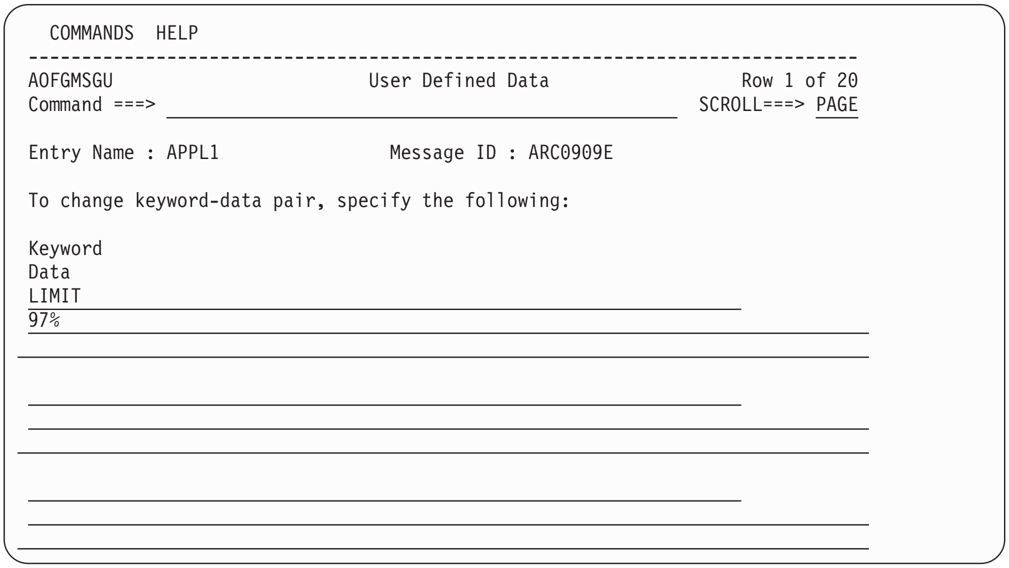
Code 1	Code 2	Code 3	Value Returned
ERROR*	$PJF		STOPPING
ABEND*	S222		ABENDING HOT

A match occurs with the second code definition, and the application is placed in the status ABENDING and the start type for the next application startup is set to HOT.

More information about this panel is given in the online help.

User Defined Data

SA z/OS displays the User Defined Data panel, as shown in Figure 121, if you entered the action code USR in the **Action** column on the Message Processing panel.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGU                        User Defined Data                 Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Name : APPL1                Message ID : ARC0909E

To change keyword-data pair, specify the following:

Keyword
Data
LIMIT
97%
```

Figure 121. User Defined Data Panel

You can use this panel to let SA z/OS perform user specified actions in case the current message requires more specific automation than provided by the generic routines. Using the ACFFQRY common routine from within a CLIST, you can retrieve the keywords entered into this panel and process them as needed. Syntax rules for how to specify user data is available in the online help. For information on how to use ACFFQRY in a CLIST refer to ACFFQRY in *IBM Tivoli System Automation for z/OS Programmer's Reference*.

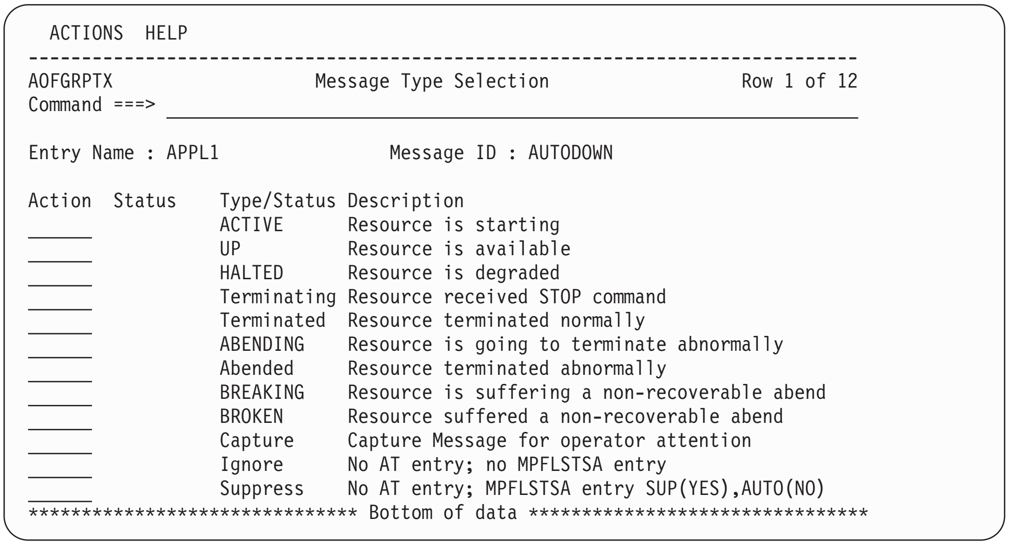
The following fields are available:

Keyword
A value is required and should represent the keyword that will be assigned the User Data.

Data A value is required and can contain any characters. There are, however, rules regarding how the data value must be delimited if it contains any special characters. For more information about these rules, see the online help.

AUTO Processing

If you enter AUT in the **Action** field on the Message Processing panel, this leads to the Message Type Selection panel, as shown in Figure 122 on page 189.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGRPTX                  Message Type Selection                 Row 1 of 12
Command ===> _________________________________________________________________

Entry Name : APPL1               Message ID : AUTODOWN

Action  Status    Type/Status Description
______            ACTIVE      Resource is starting
______            UP          Resource is available
______            HALTED      Resource is degraded
______            Terminating Resource received STOP command
______            Terminated  Resource terminated normally
______            ABENDING    Resource is going to terminate abnormally
______            Abended     Resource terminated abnormally
______            BREAKING    Resource is suffering a non-recoverable abend
______            BROKEN      Resource suffered a non-recoverable abend
______            Capture     Capture Message for operator attention
______            Ignore      No AT entry; no MPFLSTSA entry
______            Suppress    No AT entry; MPFLSTSA entry SUP(YES),AUTO(NO)
****************************** Bottom of data ********************************
```

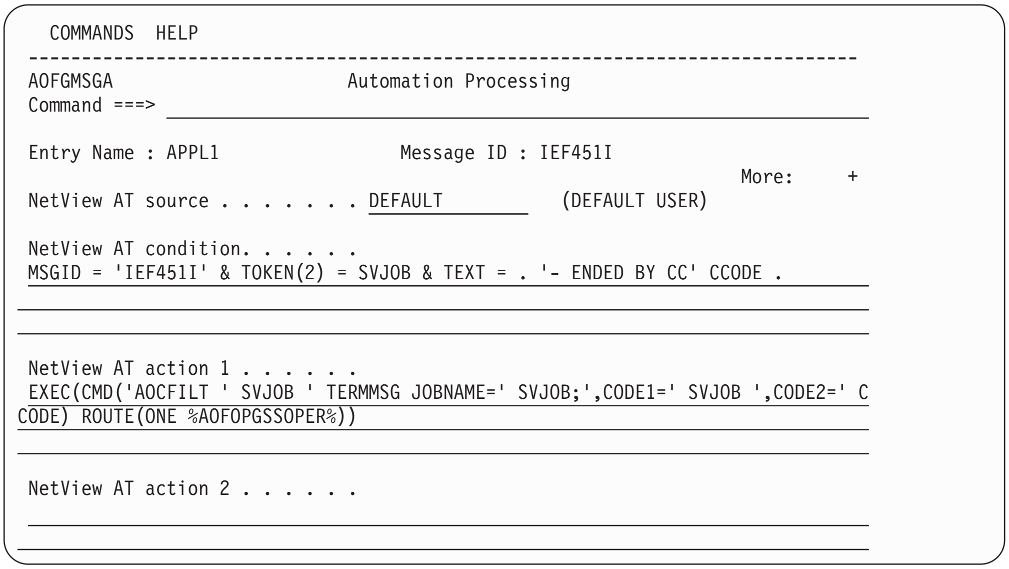
Figure 122. Message Type Selection Panel

On this panel you can select a message type or status that is associated with the message ID.

Note: Your selection influences the automation table build. Use the OVR action to verify how the automation table statement is generated.

Overriding Message Automation

If you enter OVR in the **Action** field for a message ID, the Automation Processing panel is displayed, as shown in Figure 123.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGA                     Automation Processing
Command ===> __________________________________________________________________

Entry Name : APPL1                Message ID : IEF451I
                                                                 More:     +
NetView AT source . . . . . . . DEFAULT__________  (DEFAULT USER)

NetView AT condition. . . . . .
MSGID = 'IEF451I' & TOKEN(2) = SVJOB & TEXT = . '- ENDED BY CC' CCODE .______
_______________________________________________________________________________
_______________________________________________________________________________

NetView AT action 1 . . . . . .
 EXEC(CMD('AOCFILT ' SVJOB ' TERMMSG JOBNAME=' SVJOB;',CODE1=' SVJOB ',CODE2=' C
CODE) ROUTE(ONE %AOFOPGSSOPER%))_______________________________________________
_______________________________________________________________________________

NetView AT action 2 . . . . . .
 ______________________________________________________________________________
_______________________________________________________________________________
```

Figure 123. Automation Processing Panel

This panel allows you to either define your own NetView Automation Table (AT) entries or override predefined ones for a given Entry Name and Message ID. If you override the entry, an asterisk is shown in the **OVR** column of the Message Processing panel. The Automation Processing panel has the following fields:

NetView AT source

You can set this field to:

DEFAULT
The data displayed in the **NetView AT Condition** and **Action** fields is predefined by SA z/OS. This data is used to build an AT entry unless you override it.

USER The data displayed in the **NetView AT Condition** and **Action** fields is defined by the user. This data is used to build an AT entry.

If user data exists you can toggle between user and default data by entering `USER` or `DEFAULT`.

NetView AT condition
This field displays the condition as it is used to build an AT entry. The condition field can be overridden.

This data is enclosed in IF/THEN keywords when building an AT. No IF/THEN keywords should be used in this field. A leading IF or a trailing THEN will be removed.

NetView AT action *x*
These fields display the actions as they are used to build an AT entry. All action fields can be overridden.

A semicolon ';' is appended to the last action when building an AT. No trailing semicolon should be used in this field. If used, it will be removed.

To delete user data, leave the Automation Processing panel using the END key or PF3 while either the **NetView AT source** field is set to DEFAULT or the condition field is blank.

You can view the default values by entering DEFAULT in the **NetView AT source** field. If you then try to exit the panel, a warning message appears that informs you that, by leaving the panel, you will delete your AT entry override. You can then either press the END or PF3 key to continue (and delete the entry), or press Enter to keep the override.

Notes:

1. No syntax checks will be done for the AT entry within the customization dialog.
2. Lower case characters are *not* converted to uppercase.
3. Labels defined for certain SA z/OS default AT entries will not be used if the AT entry was overridden.
4. If message automation is neither predefined by SA z/OS nor defined by the user, no AT entry is generated for the given Entry Name.

Nested Class Details

CMD (Command), REP (Reply), COD (Code), and USR (User Data) policies are inherited per message ID. Thus if a message ID has a command definition for an instance and the same message ID is defined for a class with reply data, the command and reply data are not merged on the instance, and the class definitions are not inherited at all.

Message Override and Status specifications provide instructions for the generation of the NetView automation table entry. This data is never inherited, but is used to create one AT entry for the object where they are specified.

Consider the following examples that demonstrate how policy values are inherited and AT entries are built:

No overrides

A class has a command and a linked instance has a reply defined for the message ID, ABC123.

Result: The following AT entry is built:

```
IF
MSGID = 'ABC123'
 THEN
 EXEC(CMD('ISSUEACT ')ROUTE(ONE %AOFOPGSSOPER%));
```

However, the command is *not* executed because it is not inherited by the instance and so is not loaded during automation agent initialization.

Class override

A class has a command and an override defined for a message ID, and a linked instance has a reply defined for the same message ID.

Result: Two AT entries are generated: one for the override defined on the class and one generic entry triggered by the reply defined for the instance.

Instance override

A class has a command defined for a message ID, and a linked instance has a reply and an override defined for the same message ID.

Result: Two AT entries are generated: one for the override defined on the instance and one generic entry triggered by the reply defined on the instance.

Class and instance override

A class has a command and an override defined for a message ID, and a linked instance has a reply and an override defined for the same message ID.

Result: Two AT entries are generated: one for the override defined for the class and one for the instance override. Note that the sequence of overrides is not defined, so with multiple overrides on different classes you cannot assume that they will appear in the same order as the class hierarchy.

Panel Interaction: The Message Processing panel shows message IDs that are defined for the current application and any data that has been inherited from linked classes. Inherited message IDs are displayed in a different color. It should be noted that Status or AT override definitions that have been specified for a class are not displayed on the panel.

If a linked class and the current application have definitions for the same message ID, only the definition for the current application is displayed.

Consider the following examples that illustrate inheritance from class message IDs that are modified, and how the panel display reflects this:

Modifying commands for an inherited message ID

Consider an inherited message ID with command and reply definitions and a status definition. If you modify a command definition, this becomes a definition for the current application along with the reply data, however the status data is not inherited by the instance.

Specifying Status for an inherited message ID

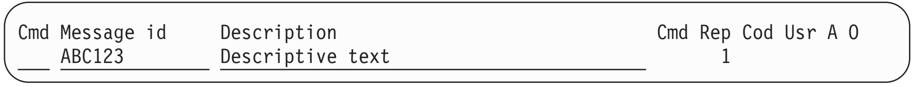
Again, consider an inherited message ID with command and reply definitions and a status definition. If you specify a status or AT override, the message ID becomes a definition for the current application, but without any data from the inherited message ID. The display shows message ID in the standard color for entries (the default is turquoise) and the counts for Cmd and Rep in the color for inherited data.

Modifying commands and adding status for an inherited message ID
Consider an inherited message ID with command and reply definitions. If you specify a status or AT override, the message ID becomes a definition for the current application with the modified commands and the reply definitions from the inherited message ID. The complete record is shown in turquoise.

Modifying just an inherited message ID or its description
Consider an inherited message ID with only a status definition. Only the message ID and its description are displayed; all other fields are blank. If you change only the message ID or description, it becomes a definition for the current application (and is redisplayed in turquoise without further data).

Status on a class
Consider a class definition that has a message ID with a Status definition, for example UP, and a linked instance that has a command defined for the same message ID.

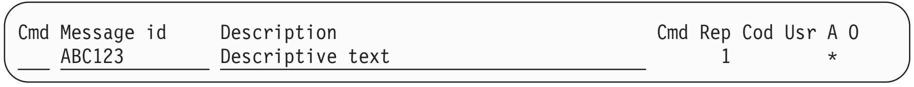
One AT entry is generated using ACTIVMSG that also runs the command specified for the instance. The following line is displayed all in turquoise:



```
Cmd Message id     Description                               Cmd Rep Cod Usr A O
___ ABC123         Descriptive text                                1
```

Status on the instance
Consider a class definition that has a command defined for message ID, and a linked instance that has a reply defined for the same message ID and also a Status definition, for example BROKEN.

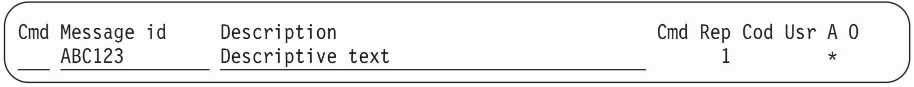
One AT entry is generated using TERMMSG, the reply from the instance is executed, but the command from the class is not inherited (and thus not executed). The following line is displayed all in turquoise:



```
Cmd Message id     Description                               Cmd Rep Cod Usr A O
___ ABC123         Descriptive text                                1         *
```

Override and Status
Consider a class definition that has a command defined for message ID and also an override, and a linked instance that has a reply defined for the same message ID and also a Status definition.

Two AT entries are generated: one for the override that is defined for the class and one for the Status specification for the instance. The following line is displayed all in turquoise:



```
Cmd Message id     Description                               Cmd Rep Cod Usr A O
___ ABC123         Descriptive text                                1         *
```

Extended Status Command Support

The status command concept has been extended so that commands can be issued if another application reaches an “up” or “down” state. Thus, a command can be issued for one application (APL1) while another one (APL2) enters a certain state.

These two applications are *consumers* that consume services provided by a *provider*. In certain cases it is valuable to trigger an action for the consumer if the provider enters an UP or DOWN state.

For further details about extended status commands, see the section "Status Commands" in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

In addition to runtime variables like &SUBSAPPL or &SUBSJOB, there are provider-specific runtime variables that can be used within a command or reply as specified for a Message ID in the MESSAGES/USER DATA policy. These variables start with &SU2 instead of &SUB. You can find a list of supported provider runtime variables in ACFCMD and ACFREP in *IBM Tivoli System Automation for z/OS Programmer's Reference*.

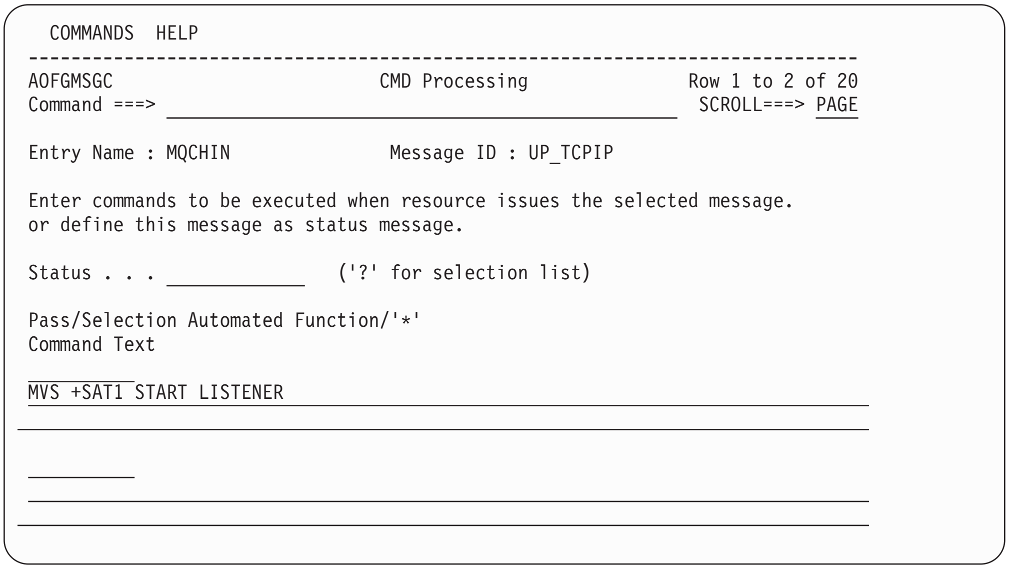
Policy Definitions: Links must be defined in the APL policy for the consumer. A link is a data pair represented by a consumer and a provider subsystem name:

- The consumer name is defined as the **Subsystem Name** of the consumer APL
- The provider name is defined in the MESSAGES/USER DATA policy item of the consumer APL as part of a Message ID

A link has two attributes that are also defined in the policy:

- The link type, which indicates whether it is a static or dynamic link:
  - A static link requires only a definition for a message ID (UP_provider-name or DN_provider-name) in the MESSAGES/USER DATA policy and an action that must be taken for the consumer
  - A dynamic link is activated at run time and therefore is declared as 'dynamic' in the policy. This is done by defining a USER action with the Keyword/Data pair DYNAMIC=YES for the message ID. There are usually multiple message IDs defined for dynamic links (one per provider) that are determined at run time
- A link condition that indicates the state of a provider when an action is taken for the consumer. The state is defined as the prefix of the message ID. It can be UP_ or DN_.

Figure 124 on page 194 shows a typical action for an UP_ message ID (in this case, starting the MQ Listener).



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGC                          CMD Processing              Row 1 to 2 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Name : MQCHIN               Message ID : UP_TCPIP

Enter commands to be executed when resource issues the selected message.
or define this message as status message.

Status . . . _____________   ('?' for selection list)

Pass/Selection Automated Function/'*'
Command Text
_________
MVS +SAT1 START LISTENER
```

Figure 124. Defining the Action for an UP_ Message ID

For dynamic links, you need to define message IDs for all the providers that might be used by a consumer. There are several alternatives for defining when a link is activated:

1. In the POSTSTART phase of the STARTUP policy item
2. Based on a user-selectable message before the consumer is in the UP state
3. Based on the “up” state (this can be UP or ENDED)
4. Based on a user-selectable message after the consumer is in the UP state

You can make these definitions in either the STARTUP or MESSAGES/USER DATA policy item. Corresponding policy definitions are required for link deactivation either in the SHUTDOWN or MESSAGES/USER DATA policy item.

You must supply a user-written REXX routine that:

1. Identifies the provider
2. Issues the INGLINK command to activate the link, if required

All the dynamic links of a consumer are automatically DEACTIVATED after the consumer is shut down. Thus you only need to define DEACTIVATE commands if you want this process to happen at an earlier point in time. Furthermore, you must not define a DEACTIVATE if the consumer is its own provider since this DEACTIVATE may take place before the DN_ message can be processed.

There is no difference in the DN_ and UP_ message processing between normal and transient applications, with one exception: when stopping a transient provider the corresponding DN_ message on the consumer is *not* processed.

Special Considerations: You can define a consumer as its own provider. This can be done to perform a predefined action for an application whenever it enters any “down” state. A “down” state in this context is an agent state of:

- DOWN
- RESTART
- AUTODOWN
- STOPPED
- CTLDOWN
- BROKEN

Thus you no longer need to define the same action for each of the agent states. To achieve this, define the same name in the DN_ message and subsystem name field of an application policy entry. You should note, however, that:

- The action defined for the DN_ message is executed even though the application is already in the "down" state. This is different to a configuration where the consumer and provider are distinct.
- Dynamic link definitions are not required. If you define a dynamic link, it will be ignored.

STARTUP Policy Item

The STARTUP policy item is used to specify scheduling subsystem, JCL procedure name (both of which are also available on the APPLICATION INFO policy item), the startup parameters and all the commands necessary to start a subsystem. These commands are issued when SA z/OS is going to start the subsystem.

The STARTUP policy item can be used to define several command sequences that are executed if an application requires certain start types that depend upon particular circumstances, such as a cold start, warm start or recovery start. You can use the INGREQ command to specify which command sequence should be taken.

You can also use this facility for performance improvements: for example (as shown in Figure 125), if application *A* depends on application *B* and *B* in turn depends on *C* and *A* also has a PREPAVAILABLE relation to *C*, after *C* is started, you can have SA z/OS issue a sequence of prestart commands to *A* while the supporting resource *B* is starting.

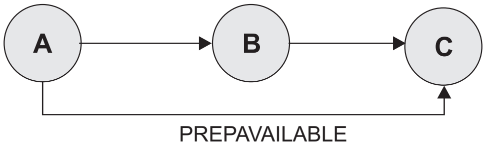
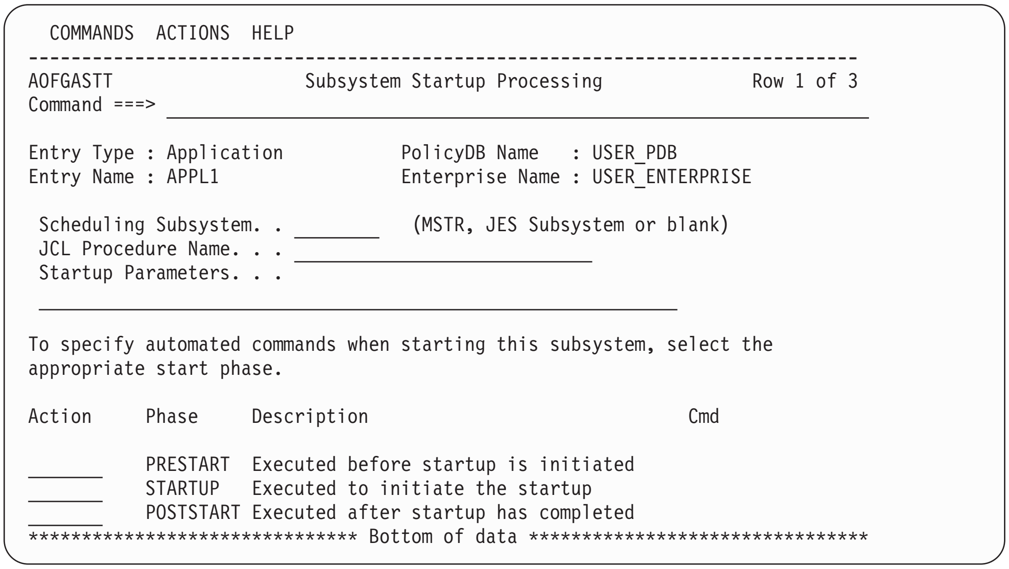


Figure 125. Using STARTUP Policy

If you select policy item STARTUP from the Policy Selection panel for Applications (Figure 108 on page 162), you come to the Subsystem Startup Processing panel shown in Figure 126 on page 196.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGASTT                 Subsystem Startup Processing             Row 1 of 3
Command ===> __________________________________________________________________

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

 Scheduling Subsystem. . ________   (MSTR, JES Subsystem or blank)
 JCL Procedure Name. . . ____________________________
 Startup Parameters. . .
 ____________________________________________________________

To specify automated commands when starting this subsystem, select the
appropriate start phase.

Action     Phase     Description                              Cmd

_______    PRESTART  Executed before startup is initiated
_______    STARTUP   Executed to initiate the startup
_______    POSTSTART Executed after startup has completed
******************************* Bottom of data ********************************
```

Figure 126. Subsystem Startup Processing Panel

For each application, you can specify a list of commands (startup procedures) that are used:

- Before the application is started
- To start the application
- After the application has started

All specified commands are processed during the different startup phases of the application. The STARTUP policy item is used to specify all the necessary commands.

The advantage of this is that you can have SA z/OS perform certain actions for the preparation of an application. For example, before starting CICS, you can have the VTAM major node activated.

The fields in the subheader area can be inherited individually from a linked class and if they have been inherited, they are shown in a different color. Any commands that have been inherited for the prestart, start or poststart phases are also shown in this color in the **Cmd** column. You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line, moving the cursor to the field of interest and pressing Enter.

Command definitions are inherited per start phase. For example, if a PRESTART command is defined for the instance and both PRESTART and STARTUP commands have been defined for the class, the instance inherits the STARTUP commands from the class.

You can fill in the following fields:

Scheduling Subsystem
This specifies the name of the subsystem that is to schedule the current application, as follows:

MSTR This indicates that the application is scheduled directly by the z/OS master scheduler.

blank This indicates that the application is scheduled by the primary JES.

other This indicates that the application is scheduled by another scheduling subsystem. The value entered is the name of that subsystem. The subsystem specified must be defined as being type JES2 or JES3 (see *Application Type* in the Define New Entry Panel for Applications Figure 107 on page 157).

This field is optional.

JCL Procedure Name

Specifies the name of the JCL procedure used to create this instance of the application. If specified, the default start command for the application will be:

```
MVS S procname,JOBNAME=jobname
```

where *procname* is the JCL procedure name and *jobname* is the job name specified above. If no JCL procedure name is specified, the default start command is:

```
MVS S jobname
```

You may include system symbols and system automation symbols. See “Assigning System Automation Symbols (AOCCLONE)” on page 27 for details on using system automation symbols.

Parameters

Specifies the desired subsystem startup parameters. These values are added onto the MVS START command for the subsystem. For example, the desired start command for VTAM may be:

```
S VTAM,,,(LIST=00)
```

Therefore, if this command is to be submitted from SA z/OS, the parameters field should contain:

```
,,,(LIST=00)
```

Whatever is entered in the parameters field is appended directly to the MVS START command for the subsystem. Therefore, the commas preceding (LIST=00) are necessary.

You may include system symbols and system automation symbols. See “Assigning System Automation Symbols (AOCCLONE)” on page 27 for details on using system automation symbols.

You may also include run time variables, such as SUBSAPPL, SUBSJOB, SUBSPROC, SUBSSHUTDLY and others. For a complete list of run time variables, refer to SA z/OS documentation.

You can specify commands for the following startup phases:

PRESTART

Commands entered for this type are issued before the application is started. You can use these to perform some preparations, for example, let SA z/OS issue VTAM commands to activate additional connections.

If a PrepAvailable is defined for the resource, the PRESTART commands are executed when the PrepAvailable condition is satisfied, or as soon as the start request becomes winning. Therefore, if the PRESTART commands rely on another resource running, a PreAvailable relationship should defined.

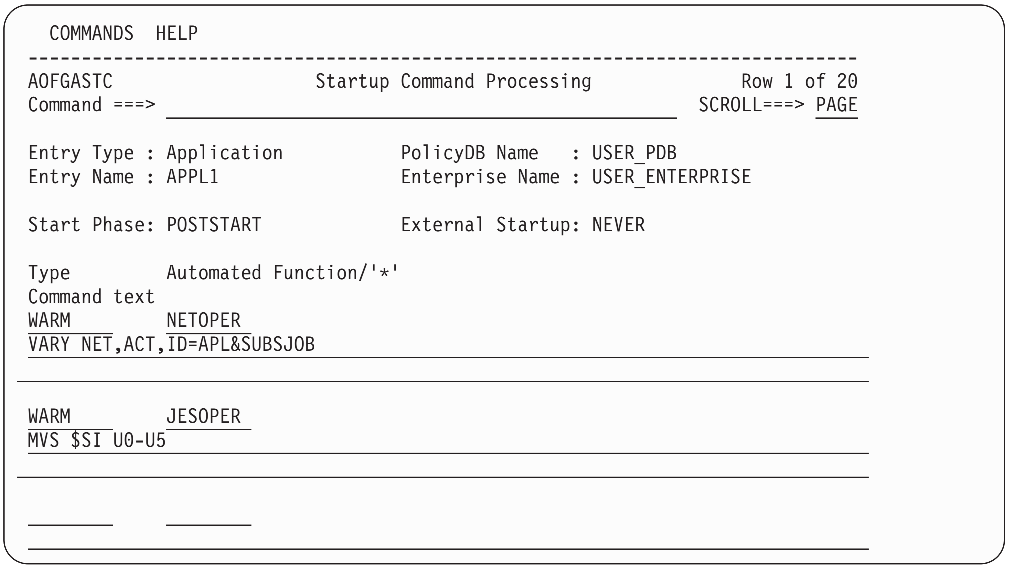
STARTUP
: You should enter only one command. This will be issued to start the application.

If you select this option, you need to specify the complete startup command, otherwise, no automatic startup command is generated from the policy specified in the *APPLICATION INFO* policy item (in entry fields *Job Name* and *JCL Procedure Name*).

POSTSTART
: Commands entered for this type are issued immediately after the UP message has been received.

The *CMD* column displays the number of commands that have been entered for a particular startup entry.

For each of these options (PRESTART, STARTUP, and POSTSTART) the Startup Command Processing panel is displayed (as shown in Figure 127), where you can enter the startup commands that are specific to the current application.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGASTC                   Startup Command Processing             Row 1 of 20
Command ===>                                                SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

Start Phase: POSTSTART             External Startup: NEVER

Type         Automated Function/'*'
Command text
WARM         NETOPER
VARY NET,ACT,ID=APL&SUBSJOB


WARM         JESOPER
MVS $SI U0-U5
```

Figure 127. Startup Command Processing Panel

This panel shows the selected start phase and the external startup value specified on policy item APPLICATION INFO and it may be used to enter the commands that are particular for this start phase.

The entries on this panel can be inherited from a class, in which case they are displayed in a different color. Because all of the commands are inherited together, if you edit one of them they all change color to indicate that they are now particular to that instance.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line.

It contains the following fields:

External Startup
: This shows whether the application is started externally or via the specified startup commands.

Type Enter the condition (for example, COLD) that the command should be executed for.

NORM is a predefined type of SA z/OS.

Leaving this field blank means that the command is executed for any condition.

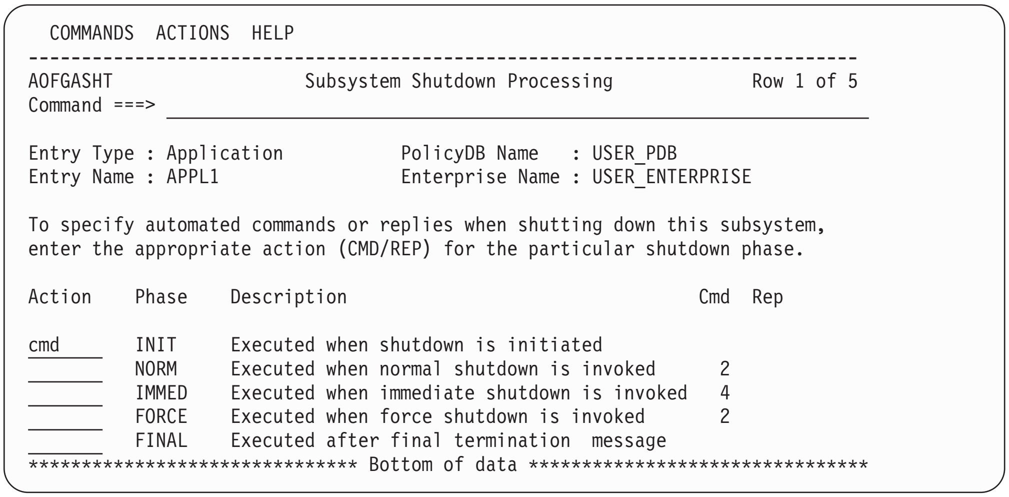
Automated Function/'*'
Enter the name of the automated function that the command is to run under. If the field is left blank, the function defaults to the one currently running the command list. SA z/OS automated functions are mapped to NetView automation operators (see Table 7 on page 292 for more details).

Note:
Entering an '*' enables return code checking for the specified command. This means, processing continues if the command returns a return code of 0. In all other cases the processing is stopped immediately. If running in LOG assist mode, return-code checking has no effect because the commands are not executed.

Command Text
Enter the command or command list to be issued. Any system command must be prefaced with MVS.

SHUTDOWN Policy Item

If you select the SHUTDOWN policy item from the Policy Selection panel for Applications, the Subsystem Shutdown Processing panel is displayed, sa shown in Figure 128.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGASHT                  Subsystem Shutdown Processing             Row 1 of 5
Command ===> __________________________________________________________________

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

To specify automated commands or replies when shutting down this subsystem,
enter the appropriate action (CMD/REP) for the particular shutdown phase.

Action    Phase    Description                                  Cmd  Rep

cmd____   INIT     Executed when shutdown is initiated
_______   NORM     Executed when normal shutdown is invoked       2
_______   IMMED    Executed when immediate shutdown is invoked    4
_______   FORCE    Executed when force shutdown is invoked        2
_______   FINAL    Executed after final termination  message
****************************** Bottom of data ********************************
```

Figure 128. Subsystem Shutdown Processing Panel

The SHUTDOWN option is used to specify all the commands necessary to shut down a subsystem and follow-on activities.

Any commands or replies that have been inherited from linked classes for the different shutdown phases are shown in a different color in the **Cmd** and **Rep** columns.

You should bear in mind the following inheritance behavior for shutdown definitions:

- Shutdown specifications are inherited per phase. For example, if you define a SHUTINIT command for an instance that has been linked to a class that has both SHUTINIT and SHUTNORM commands defined for it, the instance inherits only the SHUTNORM commands from the class.
- Command and reply definitions for each phase are inherited together. For example, if you define a command for the SHUTFORCE phase for an instance that is linked to a class that has a reply defined for SHUTFORCE, nothing is inherited by that instance.
- If you modify any inherited data, that definition becomes one for the current application. For example, if command and reply definitions are inherited for a phase and you then modify the commands, both the reply and command definitions become data for the current application. This is also the case if an instance inherits only command definitions for a phase and you then define a reply for that phase.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line, moving the cursor to the field of interest and pressing Enter.

Because shutdown procedures may involve replies to outstanding messages as well as commands, the **Action** field is used to indicate whether a command or reply is to be specified for a particular phase of shutdown. If you enter either CMD or REP, you go to another panel where the actual commands and replies can be defined, as shown in Figure 129 on page 202, Figure 130 on page 203, and Figure 132 on page 206.

Similar to the startup processing of applications described in the previous section, the following commands or replies can be chosen from this panel:

INIT Commands and replies entered under this phase are issued when the shutdown process is initiated (before the subsystem is brought down). Possible entries could be commands to send out warning messages to users or quiescing activities.

If you specified a PREPUNAVAILABLE relationship, the commands listed under INIT will be executed if the condition is satisfied.

Initial shutdown commands do any necessary processing for an application when shutdown begins. For example, you can use initial shutdown commands to issue a warning to any logged-on users that the application will be shutting down. Or, for the JES2 subsystem, you may wish to stop users on JES2 NJE lines from starting new transmissions or sessions.

The initial shutdown commands should not be used to actually shut the application down as unpredictable results may occur.

NORM

If the NORM option is selected on the INGREQ STOP command, all the commands and replies entered under this phase are issued. These entries should be the commands and replies that perform a normal shutdown of the subsystem.

IMMED

The commands and replies specified here are run when the IMMED option on INGREQ STOP is selected. These entries bring the subsystem down immediately.

FORCE
INGREQ STOP also has a FORCE option that runs the entries specified here if selected. The commands and replies would be the typical ones a user would enter to force a subsystem off the system without any delay.

FINAL
The commands specified here are run after the final termination message has been received.

Note: Shutdown reply processing is not allowed for FINAL phases, that is, you can only specify commands for this shutdown phase.

The **Cmd** and **Rep** columns cannot be filled in, but instead display the number of commands or replies that have been entered or inherited from a linked class for a particular shutdown entry.

You can use the multiple pass feature to issue one or more groups of shutdown commands or replies for all shutdown types except initial shutdown. Commands or replies with 1 in their corresponding pass fields are the entries that are issued during the first attempt at shutting down the subsystem. If after a specified amount of time (entered in the Shut Delay parameter of the Application Automation Definition panel) the subsystem is not yet down, another attempt (or pass) is tried. The next pass runs all entries that have 2 in their pass field. This continues until the subsystem is down. Each pass may consist of zero or more commands and zero or one reply. Note that issuing multiple replies on a single pass is NOT advisable. Either code secondary replies on subsequent passes or chain them through the automation table and ISSUEREP.

All commands for a given pass are processed before any replies for that pass are processed. After each set of commands and replies is issued, SA z/OS will wait for the amount of time defined in the Shut Delay field on the Application Automation Definition panel. After that time, if a call to the application monitor routine shows that the application has not shut down, the commands or reply, or both, defined for the next pass will be issued. If a call to the application monitor routine shows that the application has shut down, any subsequent shutdown passes will not be processed. If the final termination message is not received within the time specified in AOFSHUTDELAY, SA z/OS assumes that the application is down.

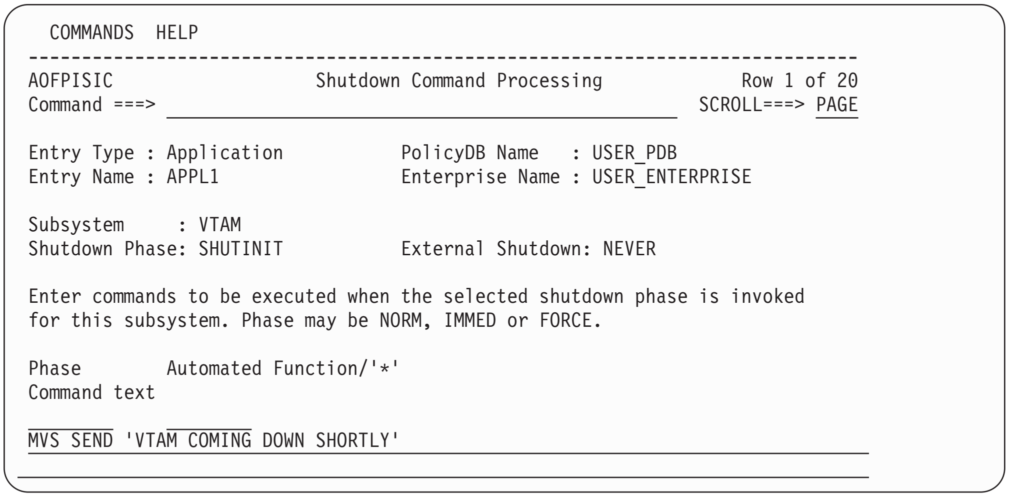
If SA z/OS processes all the shutdown passes defined for an application and the application monitor routine still shows the application as active, a warning is sent to the operator and the application status is set to STUCK.

The next panel displayed depends on the shutdown processing that you specify for the shutdown phases (INIT, NORM, etc.) on the Subsystem Shutdown Processing panel. You can specify either command processing CMD (C) or reply processing REP (R) for all shutdown types (except for FINAL, where you can specify only CMD). How to proceed on these follow-on panels is described in the following sections:

- “Command Shutdown Processing for INIT and FINAL” on page 202
- “Command Shutdown Processing for NORM, IMMED, and FORCE” on page 203
- “Reply Shutdown Processing For INIT” on page 204
- “Reply Shutdown Processing For NORM, IMMED, and FORCE” on page 205

Command Shutdown Processing for INIT and FINAL

For INIT and FINAL shutdown commands, the Shutdown Command Processing panel is displayed, as shown in Figure 129.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISIC                Shutdown Command Processing            Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : APPL1                Enterprise Name : USER_ENTERPRISE

Subsystem     : VTAM
Shutdown Phase: SHUTINIT          External Shutdown: NEVER

Enter commands to be executed when the selected shutdown phase is invoked
for this subsystem. Phase may be NORM, IMMED or FORCE.

Phase       Automated Function/'*'
Command text

________    ________
MVS SEND 'VTAM COMING DOWN SHORTLY'________________________________________
_______________________________________________________________________________
```

Figure 129. Shutdown Command Processing Panel for INIT and FINAL

This panel shows the subsystem name and the external startup value specified on the Application Info policy. It may be used to enter the commands that are particular for the specified shutdown phase, as follows:

External Shutdown
This shows whether the application is stopped externally or via the specified shutdown commands.

The following fields can be filled in:

Phase Enter the condition that the command should be executed for. NORM, IMMED and FORCE are allowed types and correspond to the SHUTNORM, SHUTIMMED and SHUTFORCE shutdown phases respectively.

Automated Function
Enter the name of the automated function that the command is to run under. If the field is left blank, the function defaults to the one currently running the command list.

> **Note:**
> Entering a ′*′ enables return code checking for the specified command. This means that processing continues if the command returns a return code of 0. In all other cases processing is stopped immediately. If running in LOG assist mode, return-code checking has no effect because the commands are not executed.

Command Text
Enter the command or command list to be issued. Any system command must be prefaced with MVS.

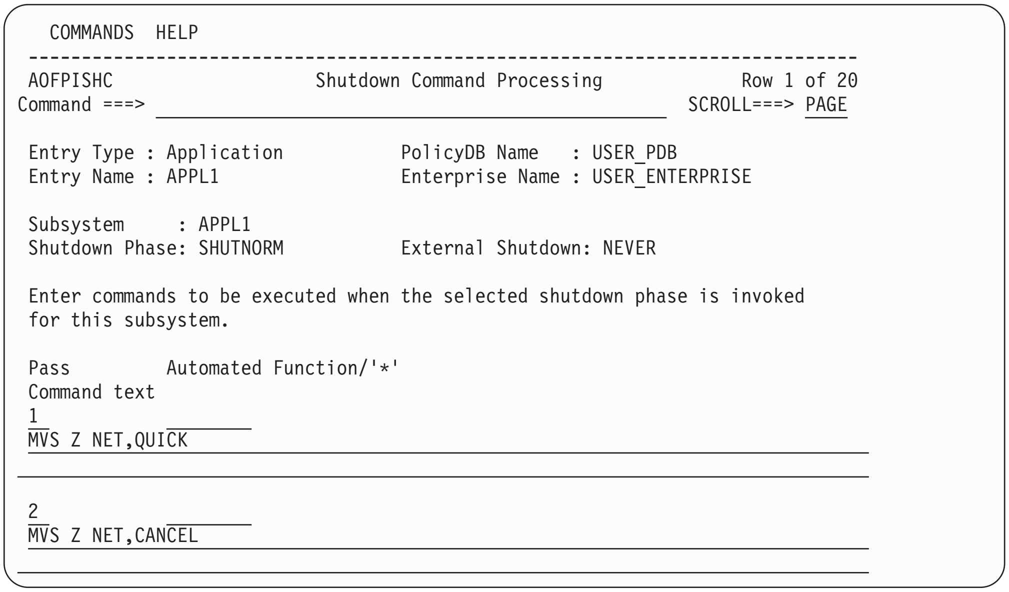
You cannot issue multiple passes of initial shutdown commands. Do not invoke MDFYSHUT.

The entries on this panel can be inherited from a class, in which case they are displayed in a different color. Because all of the commands are inherited together, if you edit one of them they all change color to indicate that they are now particular to that instance.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line.

Command Shutdown Processing for NORM, IMMED, and FORCE

For NORM, IMMED, and FORCE shutdown commands, the Shutdown Command Processing panel is displayed, as shown in Figure 130.



```
   COMMANDS  HELP
 ------------------------------------------------------------------------------
 AOFPISHC                  Shutdown Command Processing             Row 1 of 20
Command ===> ______________________________________________ SCROLL===> PAGE

 Entry Type : Application           PolicyDB Name   : USER_PDB
 Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

 Subsystem     : APPL1
 Shutdown Phase: SHUTNORM           External Shutdown: NEVER

 Enter commands to be executed when the selected shutdown phase is invoked
 for this subsystem.

 Pass         Automated Function/'*'
 Command text
 1            ________
 MVS Z NET,QUICK_______________________________________________________________
_______________________________________________________________________________

 2            ________
 MVS Z NET,CANCEL______________________________________________________________
_______________________________________________________________________________
```

Figure 130. Shutdown Command Processing Panel for NORM, IMMED, and FORCE

This panel shows the subsystem name and the external startup value specified on the Application Info policy. It may be used to enter the commands that are particular for the specified shutdown phase, as follows:

External Shutdown
This shows whether the application is stopped externally or via the specified shutdown commands.

The following fields can be filled in:

Pass Any number between 1 and 99 can be entered in this field. Commands or replies with 1 in their corresponding **Pass** field are the entries that are issued during the first attempt at shutting down the subsystem.

If after a specified amount of time (entered in the Shut Delay parameter on the Subsystem Definition panel) the subsystem is not yet down, another attempt (or pass) is tried. The next pass runs all entries that have 2 in their Pass field. This continues until the subsystem is down. If you do not specify a pass for a command, it will be issued on every pass.

For example, Figure 130 defines the normal shutdown commands issued for the VTAM subsystem. This example defines two passes of shutdown commands. On the first pass, the command `MVS Z NET,QUICK` is issued. On the second pass, the command `MVS Z NET,CANCEL` is issued.

In another example, `MVS P TSO` could be a command for pass 1 and `MVS C TSO` could be the command for pass 2 (which means that if purging TSO fails, it is canceled).

Automated Function
Enter the name of the automated function that the command is to run under. If the field is left blank, the function defaults to the one currently running the command list.

> **Note:**
> Entering a ’*’ enables return code checking for the specified command. This means that processing continues if the command returns a return code of 0. In all other cases processing is stopped immediately. If running in LOG assist mode, return-code checking has no effect because the commands are not executed.

Command Text
Enter the command or command list to be issued. Any system command must be prefaced with MVS.

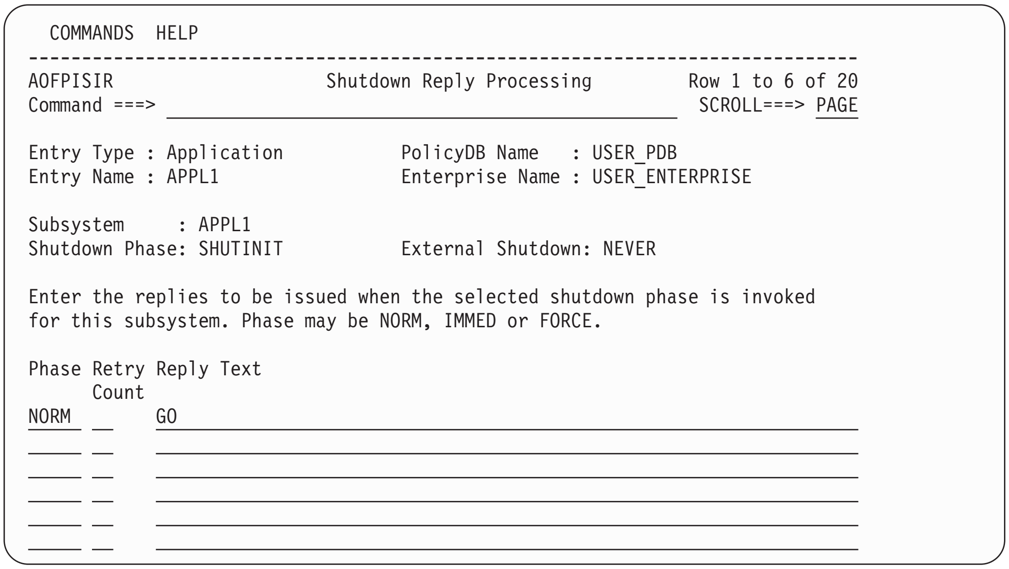
If you want to use MDFYSHUT to change the shutdown pass processing, you should specify it as a command on one of the passes.

The entries on this panel can be inherited from a class, in which case they are displayed in a different color. Because all of the commands are inherited together, if you edit one of them they all change color to indicate that they are now particular to that instance.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line.

Reply Shutdown Processing For INIT

For INIT shutdown replies, the Shutdown Reply Processing panel is displayed as shown in Figure 131.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISIR                   Shutdown Reply Processing        Row 1 to 6 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : APPL1                Enterprise Name : USER_ENTERPRISE

Subsystem     : APPL1
Shutdown Phase: SHUTINIT          External Shutdown: NEVER

Enter the replies to be issued when the selected shutdown phase is invoked
for this subsystem. Phase may be NORM, IMMED or FORCE.

Phase Retry Reply Text
      Count
NORM_ __    GO________________________________________________________________
_____ __    __________________________________________________________________
_____ __    __________________________________________________________________
_____ __    __________________________________________________________________
_____ __    __________________________________________________________________
_____ __    __________________________________________________________________
```

Figure 131. Shutdown Reply Processing Panel for INIT

This panel shows the subsystem name and the external startup value that was specified in the Application Info policy. Use it to enter the replies that are particular to the specified shutdown phase, as follows:

External Shutdown
This shows whether the application is stopped externally or via the specified shutdown commands.

The following fields can be filled in:

Phase Enter the condition that the command should be executed for. The following values and their corresponding shutdown phases are allowed:

Condition	Shutdown Phase
NORM	SHUTNORM
IMMED	SHUTIMMED
FORCE	SHUTFORCE

Retry Count
A number in the range 1 to 99 can be entered in this field to define the retry attempts made before the common routine decides that a reply cannot be issued. A retry occurs when the outstanding reply number is not available. For the first retry there is a one-second delay, and for all further retries there is a two-second delay. The default retry count is 0.

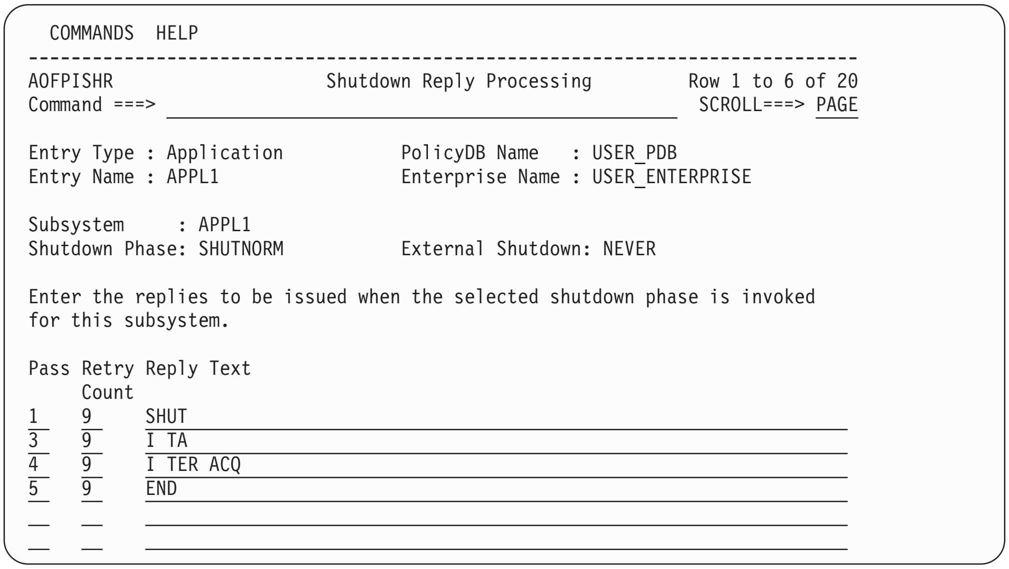
Reply Text
Enter the reply to be issued when the selected shutdown phase is invoked for this subsystem. For a null reply enter NULL as the reply.

The entries on this panel can be inherited from a class, in which case they are displayed in a different color. Because all of the commands are inherited together, if you edit one of them they all change color to indicate that they are now particular to that instance.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line.

Reply Shutdown Processing For NORM, IMMED, and FORCE

For NORM, IMMED, and FORCE shutdown replies, the Shutdown Reply Processing panel is displayed, as shown in Figure 132 on page 206.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISHR                  Shutdown Reply Processing        Row 1 to 6 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

Subsystem     : APPL1
Shutdown Phase: SHUTNORM           External Shutdown: NEVER

Enter the replies to be issued when the selected shutdown phase is invoked
for this subsystem.

Pass Retry Reply Text
     Count
1    9     SHUT
3    9     I TA
4    9     I TER ACQ
5    9     END
__   __    ______________________________________________________________
__   __    ______________________________________________________________
```

Figure 132. Shutdown Reply Processing Panel for NORM, IMMED and FORCE

This panel shows the subsystem name and the external startup value that was specified in the Application Info policy. Use it to enter the replies that are particular to the specified shutdown phase, as follows:

External Shutdown
This shows whether the application is stopped externally or via the specified shutdown commands.

The following fields can be filled in:

Pass
Any number between 1 and 99 can be entered in this field. Commands or replies with 1 in their corresponding Pass field are the entries that are issued during the first attempt at shutting down the subsystem.

If after a specified amount of time (entered in the Shut Delay parameter from Subsystem Definition panel) the subsystem is not yet down, another attempt (or pass) is tried. The next pass runs all entries that have 2 in their Pass field. This continues until the subsystem is down. A pass number is required for each shutdown reply.

For example, `MVS P TSO` could be a command for pass 1 and `MVS C TSO` could be the command for pass 2 (which means that if purging TSO fails, it is canceled).

Retry Count
A number in the range 1 through 99 can be entered in this field to define the retry attempts made before the common routine decides that a reply cannot be issued. A retry occurs when the outstanding reply number is not available. For the first retry there is a one-second delay, and for all further retries there is a two-second delay. The default retry count is 0.

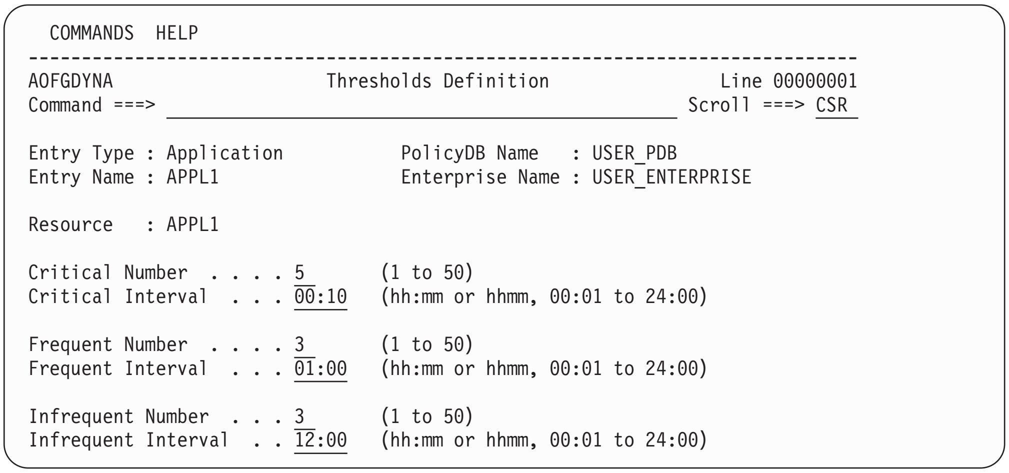
Reply Text
Enter the reply to be issued when the selected shutdown phase is invoked for this subsystem. For a null reply enter NULL as the reply.

The entries on this panel can be inherited from a class, in which case they are displayed in a different color. Because all of the commands are inherited together, if you edit one of them they all change color to indicate that they are now particular to that instance.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line.

THRESHOLDS Policy Item

The Thresholds Definition panel is displayed if you select policy item THRESHOLDS on the Policy Selection panel for applications, as shown in Figure 133.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA                   Thresholds Definition                 Line 00000001
Command ===>                                               Scroll ===> CSR

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

Resource   : APPL1

Critical Number  . . . . 5       (1 to 50)
Critical Interval  . . . 00:10   (hh:mm or hhmm, 00:01 to 24:00)

Frequent Number  . . . . 3       (1 to 50)
Frequent Interval  . . . 01:00   (hh:mm or hhmm, 00:01 to 24:00)

Infrequent Number  . . . 3       (1 to 50)
Infrequent Interval  . . 12:00   (hh:mm or hhmm, 00:01 to 24:00)
```

Figure 133. Thresholds Definition Panel

The information provided in the THRESHOLDS dialogs is used to determine when an operator should be notified of an error situation. An attempt at recovering an error (such as an abend of a subsystem) is handled by automation, but an operator may want to know if this error is occurring often.

Thresholds can be set that let the operator know if certain errors are occurring infrequently, frequently or have reached a critical stage where the recovery process should be ended (to avoid endless loops). This is done by specifying how many times an error must happen in a certain time period for each error situation.

The entries on this panel can be inherited from a class, in which case they are displayed in a different color. Because all of the thresholds are inherited together, if you edit one of them, all of the thresholds change color to indicate that they are now particular to that instance because they are no longer inherited.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line.

You can define three different threshold levels:

Infrequent
Issues a warning message and restart processing is done.

Frequent
Issues a warning message, but restart processing is still done.

Critical
Sets the resource to a status of BROKEN, stops the restart automation and issues a message.

You can enter data in the following fields:

Number
: This field is used to define the number of occurrences of an error that creates the particular error level. The range of allowable numbers is 1 to 50.

Interval
: The number of errors entered in the previous field must occur in a certain time period in order to create a specific error level. This time period is placed in this field. The format of this field is in hours and minutes. The range for hours is 00 to 24 and minutes 00 to 59. The colon between hours and minutes is optional (can be entered as HH:MM or HHMM).

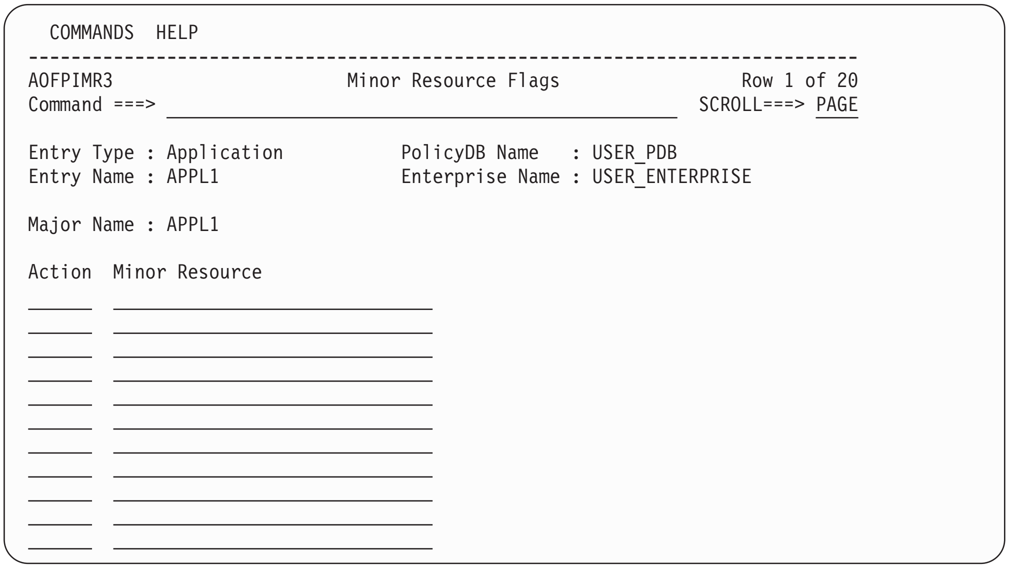
The values entered in the **Number** and **Interval** fields should be such that the threshold levels are reached in a logical manner (a critical threshold should not be reached before a frequent threshold).

Not all three threshold levels need be specified, for example, if an application needs only a specification for the critical number and interval, you can leave the frequent and infrequent fields empty.

For more information on setting thresholds refer to “How SA z/OS Uses Error Thresholds” on page 17.

MINOR RESOURCE FLAGS Policy Item

The Minor Resource Selection panel, as shown in Figure 134, is displayed if you select policy item MINOR RESOURCE FLAGS on the Policy Selection panel for applications.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIMR3                       Minor Resource Flags                 Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

Major Name : APPL1

Action  Minor Resource
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
```

Figure 134. Minor Resource Selection Panel

This panel allows you to manage automation flags for minor resources.

Major Name
: This field shows the major resource name.

Action
: Use this field to select (S) a minor resource for further processing or to delete (D) a minor resource.

Minor Resource

Use this field to specify the minor resource name. The following rules apply to the minor resource names:

- The name cannot contain any of the following characters: comma, single quotation mark, double quotation mark, left or right parenthesis, equal sign or a blank.
- The name cannot start or end with a period.
- The name cannot contain consecutive periods.

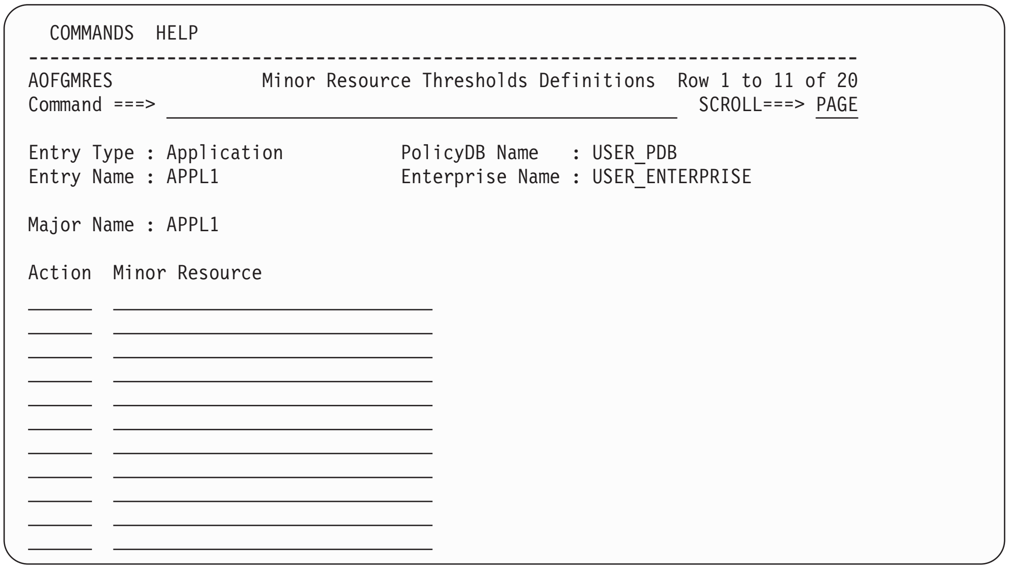
To add a minor resource enter a name in one of the empty entries at the bottom of the list.

To rename a minor resource overtype the current name with a new name and press Enter.

SA z/OS provides automation routines for recovery actions that use the automation flags that are defined for these minor resources together with commands that are defined for messages with the same name. For more information see the chapter “Automation Routines” in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

MINOR RESOURCE THRES Policy Item

The Minor Resource Thresholds Definitions panel, as shown in Figure 135, is displayed if you select the MINOR RESOURCE THRES policy item on the Policy Selection panel for applications.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMRES              Minor Resource Thresholds Definitions  Row 1 to 11 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : APPL1                 Enterprise Name : USER_ENTERPRISE

Major Name : APPL1

Action  Minor Resource
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
```

Figure 135. Minor Resource Selection Panel

This panel allows you to manage thresholds for your minor resources.

Major Name

The major resource name.

Action

Use this to select (S) a minor resource for further processing or delete (D) a minor resource.

Minor Resource

Use this to specify the minor resource name. The following rules apply to the minor resource names:

- It cannot contain the following characters: comma, single quotation mark, double quotation mark, left or right parenthesis, equals sign or a blank.
- It cannot start or end with a period.
- It cannot contain consecutive periods.

To add a minor resource, enter a name in one of the null entries at the end of the list.

To rename a minor resource overtype the current name with a new name.

For more details about minor resource thresholds, see “Thresholds for Minor Resources” on page 20.

SYSTEM ASSOCIATION Policy Item

The specified definitions are not used for automation by SA z/OS . They are for customer specific purposes only.

Using MOVED and FALLBACK

SA z/OS provides two automation statuses, MOVED and FALLBACK, to handle applications which normally run on one system (the primary) but which can run on an alternative system (the secondary) if the primary is unavailable.

SA z/OS does not provide any logic to actually perform a fallback from a primary to a secondary system. If you want cross-system fallback, you have to provide the coordinating logic yourself.

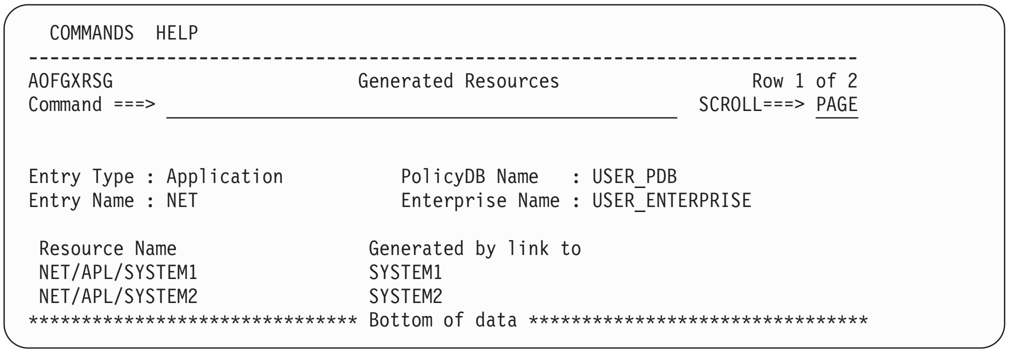
If you want the application moved automatically you will have to write your own code or define a MOVE-group. You can put applications into MOVED or FALLBACK status either through SETSTATE or AOCUPDT.

GENERATED RESOURCES Policy Item

The Generated Resources panel appears if you select policy item *GENERATED RESOURCES* from the Policy Selection panel for Applications, as shown in Figure 136 on page 211. This panel shows all resources that are generated by SA z/OS for the application. This panel is for information only. No actions are allowed.

Application Resources are generated when a link between an Application and a System occurs. Applications can never be linked directly to Systems.

This link happens indirectly by linking an Application to an Application Group (either in Application’s WHERE USED policy or in an Application Group’s APPLICATIONS policy.)



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGXRSG                       Generated Resources                 Row 1 of 2
Command ===> ________________________________________________  SCROLL===> PAGE


Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : NET                   Enterprise Name : USER_ENTERPRISE

 Resource Name                  Generated by link to
 NET/APL/SYSTEM1                SYSTEM1
 NET/APL/SYSTEM2                SYSTEM2
******************************* Bottom of data ********************************
```

Figure 136. Generated Resources Panel

This panel displays the following information:

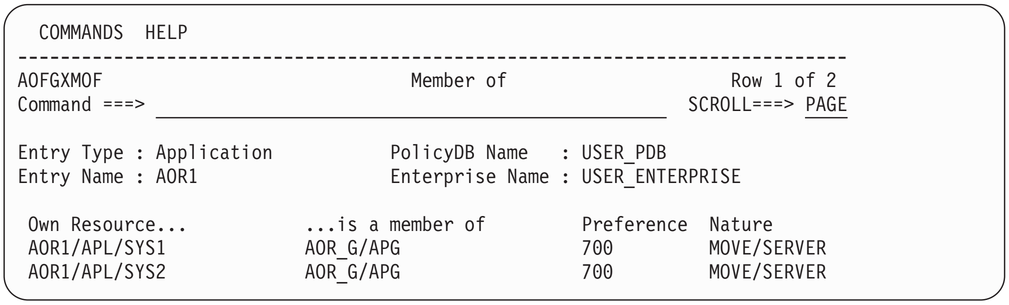
Resource Name
Shows the resources that have generated for the application.

Generated by link to
This says who the owner of a Resource is. In this case; owners are Application Groups because the current entry is an APL

A link to the owner creates the Resource, unlinking from the owner (or deletion of the owner) deletes the Resource..

MEMBER OF for Applications

The Member of panel is displayed if you select policy item *MEMBER OF* on the Policy Selection panel for applications.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGXMOF                             Member of                     Row 1 of 2
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : AOR1                  Enterprise Name : USER_ENTERPRISE

 Own Resource...          ...is a member of        Preference  Nature
 AOR1/APL/SYS1            AOR_G/APG                700         MOVE/SERVER
 AOR1/APL/SYS2            AOR_G/APG                700         MOVE/SERVER
```

Figure 137. Member of Panel for Applications

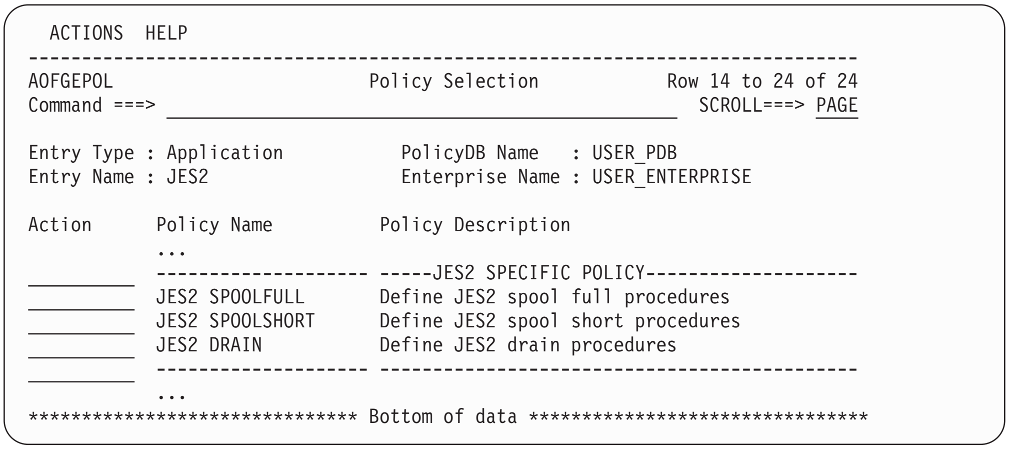
This panel lists all Resources where this Application is a member "resource-wise", that is, it lists all relations from resources that have been derived from the current Application to other resources. This panel is for information only. No data can be changed.

Use the RESOURCES policy of an APG to add or remove it to the list in the **...is a member of** column (that is in the RESOURCES policy of the APG Entry Name that the member of Resource was derived from).

In this example, this panel is invoked for application *AOR1* which is a member of a Sysplex Application Group. In such a case, you will see all resources generated for the sysplex.

Policy Items for JES2 Subsystems

Figure 138 is an example of a Policy Selection panel for a JES2 subsystem.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection            Row 14 to 24 of 24
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : JES2                  Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            ...
__________  -------------------- -----JES2 SPECIFIC POLICY--------------------
__________  JES2 SPOOLFULL       Define JES2 spool full procedures
__________  JES2 SPOOLSHORT      Define JES2 spool short procedures
__________  JES2 DRAIN           Define JES2 drain procedures
__________  -------------------- ---------------------------------------------
            ...
******************************* Bottom of data ********************************
```

Figure 138. Policy Selection Panel for JES2 Subsystems

See “Policy Items for Applications” on page 162 for descriptions of most policy items on this panel. The additional policy items are:

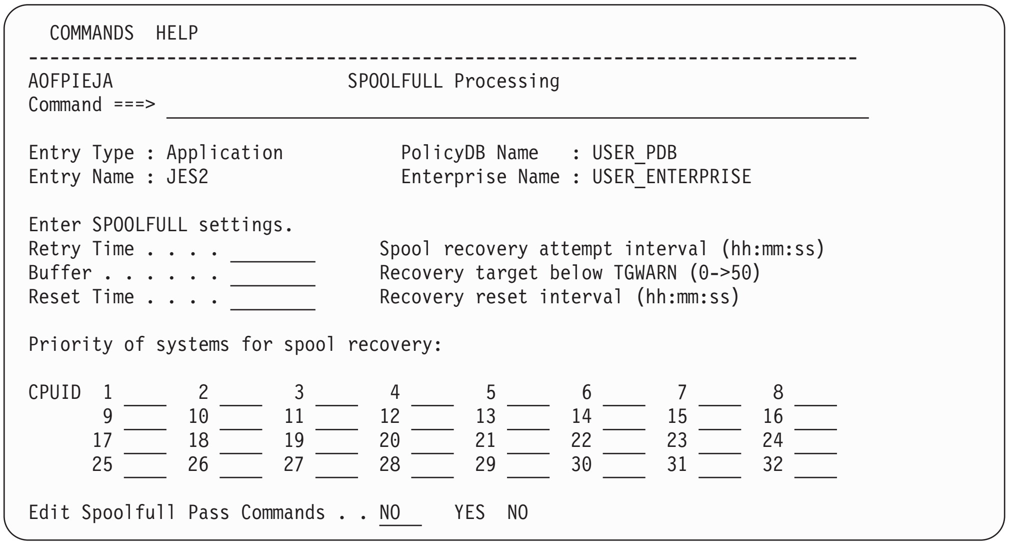
JES2 SPOOLFULL
Provides for recovery from a JES spool full condition. Figure 139 on page 213 shows the recovery parameters that can be specified for JES2 when this policy item is selected.

JES2 SPOOLSHORT
Provides for recovery from JES spool shortage condition. The recovery parameters available for JES2 spool shortage recovery are the same as for spool full recovery. Refer to “JES2 SPOOLSHORT Policy Item” on page 214.

JES2 DRAIN
Allows you to specify which JES resources are to be drained and how they are to be drained in the event that a JES2 shutdown fails.

JES2 SPOOLFULL Policy Item

If you select policy item SPOOLFULL for a JES2 subsystem, SA z/OS displays the SPOOLFULL Processing panel.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIEJA                      SPOOLFULL Processing
Command ===> __________________________________________________________________

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : JES2                  Enterprise Name : USER_ENTERPRISE

Enter SPOOLFULL settings.
Retry Time . . . . ________      Spool recovery attempt interval (hh:mm:ss)
Buffer . . . . . . ________      Recovery target below TGWARN (0->50)
Reset Time . . . . ________      Recovery reset interval (hh:mm:ss)

Priority of systems for spool recovery:

CPUID  1 ____    2 ____    3 ____    4 ____    5 ____    6 ____    7 ____    8 ____
       9 ____   10 ____   11 ____   12 ____   13 ____   14 ____   15 ____   16 ____
      17 ____   18 ____   19 ____   20 ____   21 ____   22 ____   23 ____   24 ____
      25 ____   26 ____   27 ____   28 ____   29 ____   30 ____   31 ____   32 ____

Edit Spoolfull Pass Commands . . NO     YES  NO
```

Figure 139. SPOOLFULL Processing Panel

The following parameters can be entered:

Retry Time
The time entered in this field defines the time between attempts at spool recovery. If the first set of recovery commands has been issued and the spool is not recovered after this amount of time, the second set of defined commands is issued. Time range 00:00:00 to 23:59:59. The default is 5 minutes.

Buffer The percentage amount of spool that is to be recovered under the threshold (TGWARN for SPOOLSHORT, 100% for SPOOLFULL). The value must be positive and less than 75% of the threshold value. If invalid or not specified a default of 5 is used. For example, if TGWARN is set to 80 and Buffer is set to 10, SPOOLSHORT recovery will continue until spool utilization is less than 70%.

Reset time
This is the amount of time that must elapse after the spool has been driven below the recovery target before the spool incident is considered to be closed. If the spool exceeds the trigger threshold again before this interval is up, the recovery will begin again with the next pass, rather than restarting from pass 1.

The default is 15 minutes.

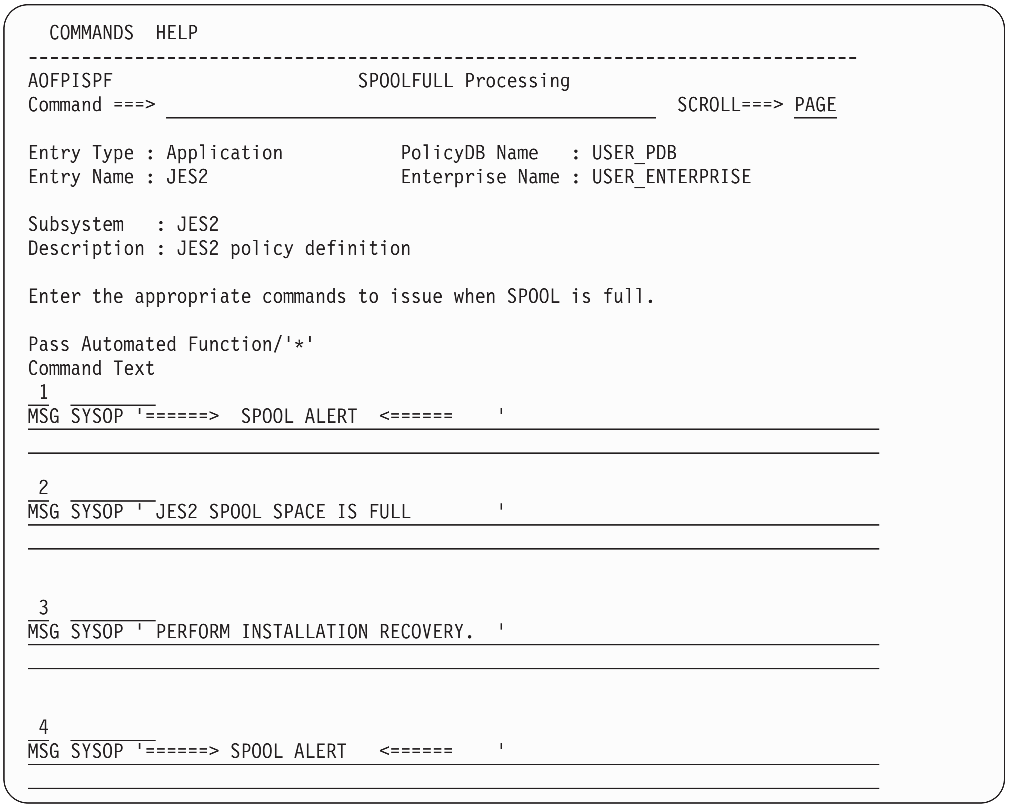
CPUID 1st - 32nd
An ordered list of system IDs belonging to the JES2 subsystems sharing the JES spool in a MAS definition. During the recovery the JES2 command D MEMBER,STATUS=ACTIVE,NAME is issued and the first machine in the list that is active on the display response is assumed to be responsible for handling the recovery.

Note:
You must keep the CPUID lists for JES subsystems sharing the spool space in synchronization.

If SA z/OS cannot find an active machine in the CPUID list, it will assume responsibility for the recovery itself. This can lead to multiple machines issuing recovery commands if your lists are incomplete.

Edit Spoolfull Pass Commands

Entering YES in this field allows you to edit the recovery commands. Figure 140 shows the SPOOLFULL Processing panel where you can enter commands to be issued during SPOOLFULL processing.



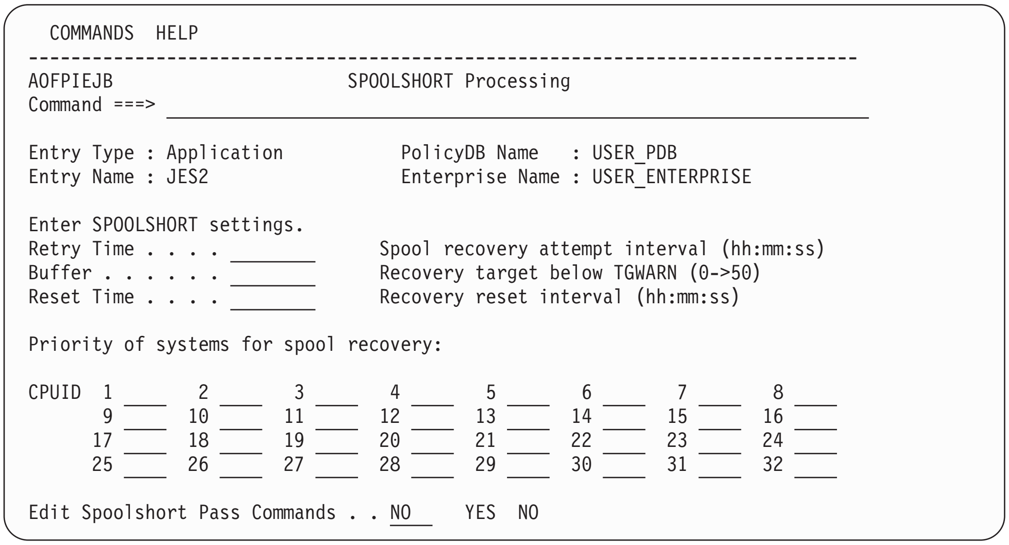
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISPF                       SPOOLFULL Processing
Command ===>                                              SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : JES2                  Enterprise Name : USER_ENTERPRISE

Subsystem   : JES2
Description : JES2 policy definition

Enter the appropriate commands to issue when SPOOL is full.

Pass Automated Function/'*'
Command Text
 1
MSG SYSOP '======>  SPOOL ALERT  <======    '


 2
MSG SYSOP ' JES2 SPOOL SPACE IS FULL        '



 3
MSG SYSOP ' PERFORM INSTALLATION RECOVERY.  '



 4
MSG SYSOP '======> SPOOL ALERT   <======    '

```

Figure 140. SPOOLFULL Processing Panel for JES2

Commands are issued by passes, so a command to be issued on every pass should be specified without pass specification.Two substitution variables (EHKVAR1 and EHKVAR2) are available. EHKVAR1 contains the current spool utilization, and EHKVAR2 contains the recovery target.

JES2 SPOOLSHORT Policy Item

If you select policy item SPOOLSHORT for a JES2 subsystem, SA z/OS displays the SPOOLSHORT Processing panel, as shown in Figure 141 on page 215.



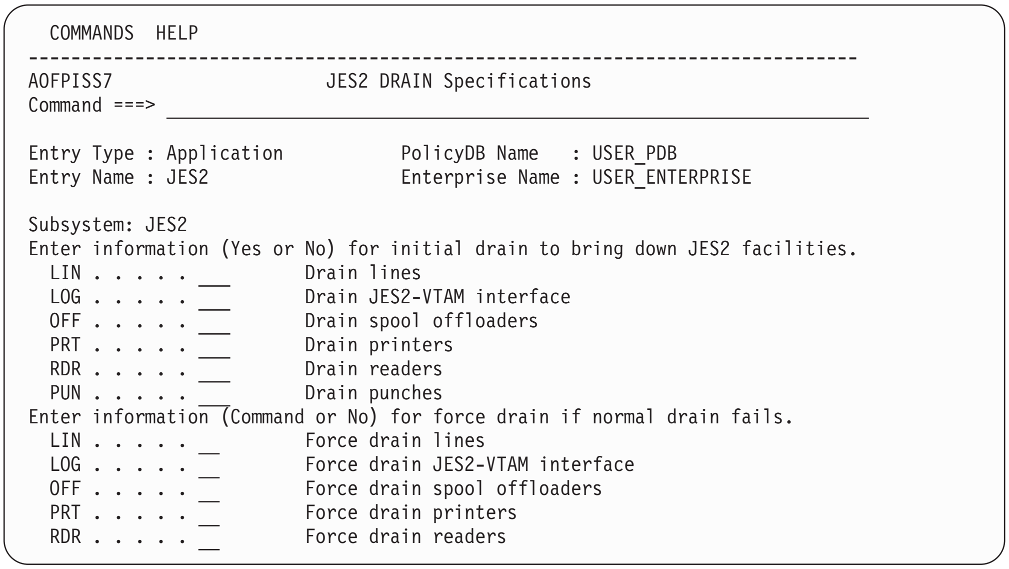
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIEJB                      SPOOLSHORT Processing
Command ===> __________________________________________________________________

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : JES2                  Enterprise Name : USER_ENTERPRISE

Enter SPOOLSHORT settings.
Retry Time . . . . ________      Spool recovery attempt interval (hh:mm:ss)
Buffer . . . . . . ________      Recovery target below TGWARN (0->50)
Reset Time . . . . ________      Recovery reset interval (hh:mm:ss)

Priority of systems for spool recovery:

CPUID  1 ____   2 ____   3 ____   4 ____   5 ____   6 ____   7 ____   8 ____
       9 ____  10 ____  11 ____  12 ____  13 ____  14 ____  15 ____  16 ____
      17 ____  18 ____  19 ____  20 ____  21 ____  22 ____  23 ____  24 ____
      25 ____  26 ____  27 ____  28 ____  29 ____  30 ____  31 ____  32 ____

Edit Spoolshort Pass Commands . . NO__   YES  NO
```

Figure 141. SPOOLSHORT Processing Panel

The parameters that can be entered on this panel are similar to those described in “JES2 SPOOLFULL Policy Item” on page 212, and the *Spoolshort Processing* panel is similar to the *Spoolfull Processing* panel.

JES2 DRAIN Policy Item

If you select policy item DRAIN for a JES2 subsystem, SA z/OS displays the JES2 DRAIN Processing panel, as shown in Figure 142.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISS7                    JES2 DRAIN Specifications
Command ===> __________________________________________________________________

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : JES2                  Enterprise Name : USER_ENTERPRISE

Subsystem: JES2
Enter information (Yes or No) for initial drain to bring down JES2 facilities.
  LIN . . . . . ___         Drain lines
  LOG . . . . . ___         Drain JES2-VTAM interface
  OFF . . . . . ___         Drain spool offloaders
  PRT . . . . . ___         Drain printers
  RDR . . . . . ___         Drain readers
  PUN . . . . . ___         Drain punches
Enter information (Command or No) for force drain if normal drain fails.
  LIN . . . . . __          Force drain lines
  LOG . . . . . __          Force drain JES2-VTAM interface
  OFF . . . . . __          Force drain spool offloaders
  PRT . . . . . __          Force drain printers
  RDR . . . . . __          Force drain readers
```

Figure 142. JES2 DRAIN Processing Panel

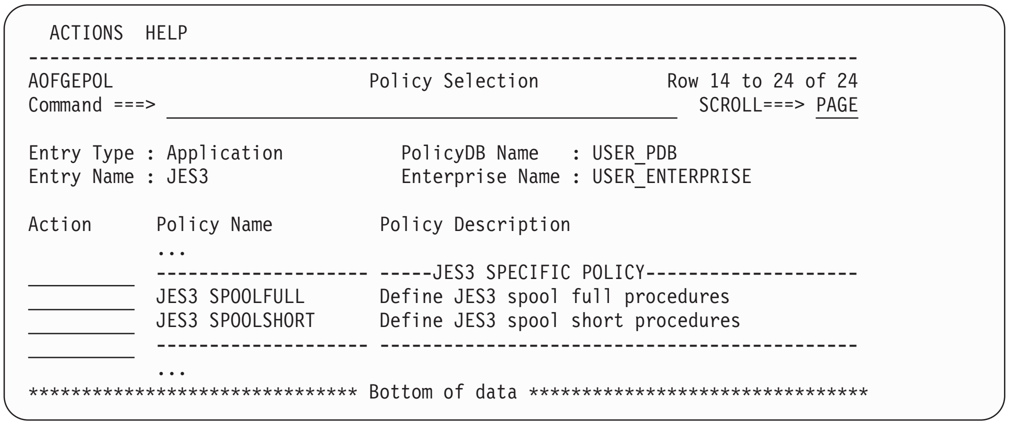
This panel is used to identify which JES resources are to be drained and how they are to be drained if a JES shutdown is unsuccessful:

Initial drain
Choose the resources that are to be stopped with a JES $P command. An initial drain occurs either because of a JES shutdown or an operator issued AOFDRAIN. YES indicates that a stop command should be issued for the resource while a NO (the default) does not drain the resource type.

Force drain
Define the drain command to be entered for each resource if JES has not successfully shutdown. Possible JES commands could be Restart ($E), Cancel ($C) or any other appropriate command. If NO is entered (the default), this resource type does not have any commands issued for it.

Policy Items for JES3 Subsystems

Figure 143 shows a Policy Selection panel for a JES3 subsystem.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection            Row 14 to 24 of 24
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : JES3                  Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            ...
__________  -------------------- -----JES3 SPECIFIC POLICY--------------------
__________  JES3 SPOOLFULL       Define JES3 spool full procedures
__________  JES3 SPOOLSHORT      Define JES3 spool short procedures
__________  -------------------- ---------------------------------------------
            ...
******************************* Bottom of data ********************************
```

Figure 143. Policy Selection Panel for JES3 Subsystems

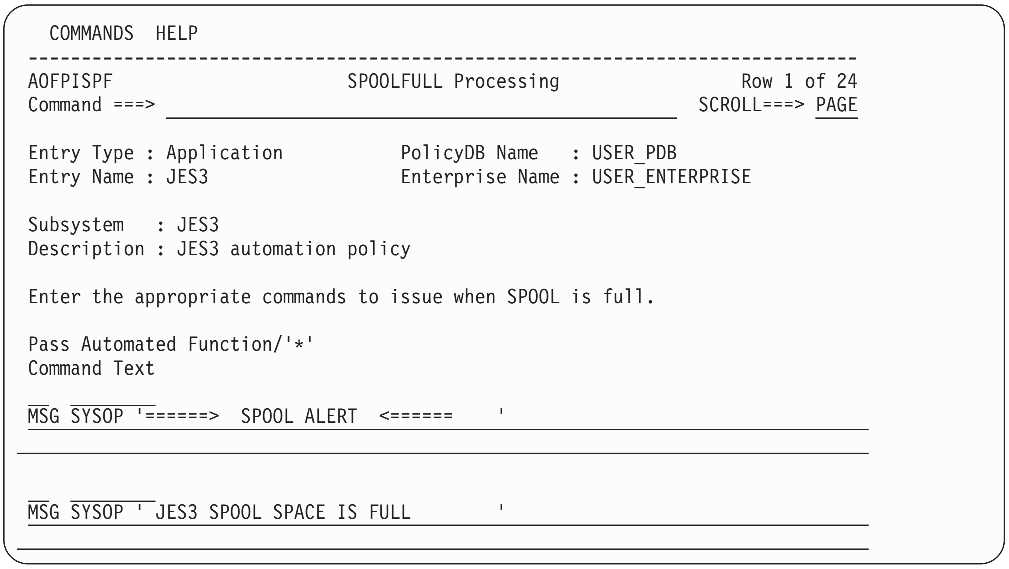
See “Policy Items for Applications” on page 162 for descriptions of most policy items on this panel. The additional policy items are:

JES3 SPOOLFULL
Allows you to specify recovery from JES spool full condition. For more information refer to “JES3 SPOOLFULL Policy Item.”

JES3 SPOOLSHORT
Allows you to specify recovery from JES spool short condition. For more information refer to “JES3 SPOOLSHORT Policy Item” on page 217.

JES3 SPOOLFULL Policy Item

If you select policy item SPOOLFULL for a JES3 subsystem, SA z/OS displays the SPOOLFULL Processing panel, as shown in Figure 144 on page 217.



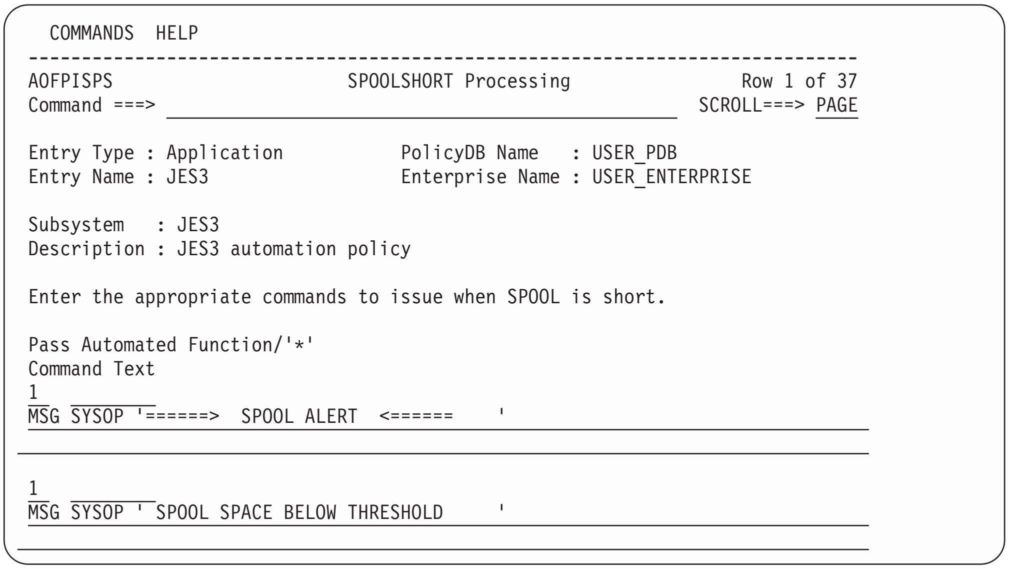
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISPF                      SPOOLFULL Processing                Row 1 of 24
Command ===>                                                  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : JES3                  Enterprise Name : USER_ENTERPRISE

Subsystem   : JES3
Description : JES3 automation policy

Enter the appropriate commands to issue when SPOOL is full.

Pass Automated Function/'*'
Command Text

MSG SYSOP '======>  SPOOL ALERT  <======    '


MSG SYSOP ' JES3 SPOOL SPACE IS FULL        '
```

Figure 144. JES3 SPOOLFULL Processing Panel

Commands are issued by passes, so a command to be issued on every pass should be specified with an asterisk * for the pass. Two substitution variables (EHKVAR1 and EHKVAR2) are available. EHKVAR1 contains the current spool utilization, and EHKVAR2 contains the recovery target.

JES3 SPOOLSHORT Policy Item

If you select policy item SPOOLSHORT for a JES3 subsystem, SA z/OS displays the SPOOLSHORT Processing panel, as shown in Figure 145.



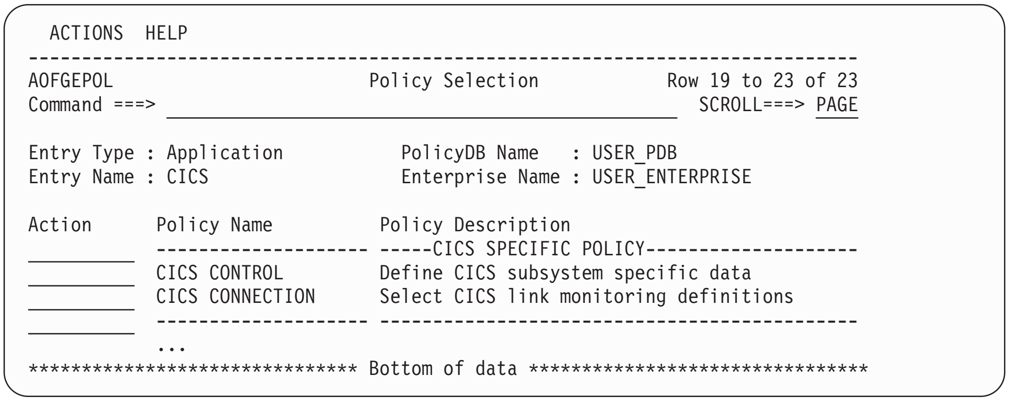
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISPS                      SPOOLSHORT Processing               Row 1 of 37
Command ===>                                                  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : JES3                  Enterprise Name : USER_ENTERPRISE

Subsystem   : JES3
Description : JES3 automation policy

Enter the appropriate commands to issue when SPOOL is short.

Pass Automated Function/'*'
Command Text
1
MSG SYSOP '======>  SPOOL ALERT  <======    '


1
MSG SYSOP ' SPOOL SPACE BELOW THRESHOLD     '
```

Figure 145. JES3 SPOOLSHORT Processing Panel

Commands are issued by passes, so a command to be issued on every pass should be specified with an asterisk * for the pass. Two substitution variables (EHKVAR1 and EHKVAR2) are available. EHKVAR1 contains the current spool utilization, and EHKVAR2 contains the recovery target.

Policy Items for CICS Subsystems

Figure 146 shows the Policy Selection panel for applications of type CICS (see the **Application Type** entry field in Figure 107 on page 157). This Policy Selection panel contains additional policy items for a CICS subsystem.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                       Policy Selection           Row 19 to 23 of 23
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : CICS                 Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            -------------------- -----CICS SPECIFIC POLICY--------------------
__________  CICS CONTROL         Define CICS subsystem specific data
__________  CICS CONNECTION      Select CICS link monitoring definitions
__________  -------------------- --------------------------------------------
__________  ...
******************************* Bottom of data ********************************
```

Figure 146. Policy Selection Panel for CICS Subsystems

The CICS specific policy items are:

CICS CONTROL
With this policy item you can define the extended subsystem information required for CICS Automation. You will reach a panel described in “CICS CONTROL Policy Item” where you can fill in the required information.

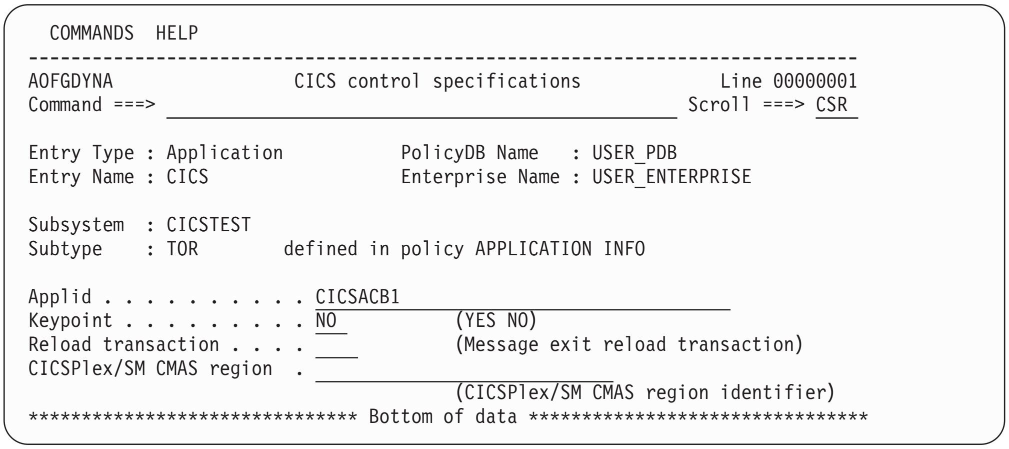
CICS CONNECTION
If you select this policy item, the panel CICS link for Application will be displayed, showing a list of CICS link definitions previously defined as described in “CICS Link Definitions Policy Object” on page 270. Use this policy item to connect a CICS link monitoring definition to the current CICS subsystem. This item is described in section CONNECTION of *IBM Tivoli System Automation for z/OS CICS Automation Programmer’s Reference and Operator’s Guide*. By selecting a link here, SA z/OS creates an entry in the automation control file as follows:

```
CONNECTION subsys.connection_id
```

where connection `connection_id` is the entry name specified when creating the entry 40/CCN. You may select multiple connections.

CICS CONTROL Policy Item

With this policy item you can define the extended subsystem information required for CICS Automation, as shown in Figure 147 on page 219.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA                CICS control specifications             Line 00000001
Command ===> ________________________________________________ Scroll ===> CSR

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : CICS                  Enterprise Name : USER_ENTERPRISE

Subsystem  : CICSTEST
Subtype    : TOR       defined in policy APPLICATION INFO

Applid . . . . . . . . . . CICSACB1
Keypoint . . . . . . . . . NO          (YES NO)
Reload transaction . . . . ____        (Message exit reload transaction)
CICSPlex/SM CMAS region  . ____________________________
                                       (CICSPlex/SM CMAS region identifier)
****************************** Bottom of data *******************************
```

Figure 147. CICS Control Specifications Panel

Subtype

This parameter indicates the resource type. It can be entered in the **Subtype** field of the APPLICATION INFO policy item of the application instance or application class. The value shown is either for the application instance or the application class that the application is linked to.

Fill in the following fields:

Applid

Use one of the following:

APPLID

Use this keyword to specify the VTAM application identifier if XRF=NO is coded on the SIT for this CICS subsystem and the name of the VTAM APPL statement and the ACB name specified in it are the same.

(VTAM APPL name,VTAM ACBNAME)

If XRF=NO is coded on the SIT for this CICS, but the name of the VTAM APPL statement is different from the ACB name specified in it, code them both in this entry. The VTAM ACBNAME must be the same as that used by the CICS application.

(*generic-applid***,***specific-applid***)**

If XRF=YES is coded on the SIT for this CICS subsystem, *generic-applid* is the name given to both the active and alternate subsystem and *specific-applid* is the specific name for this subsystem. The *specific-applid* must be the same as that used by the CICS application.

Keypoint

Determines whether a CICS keypoint is required and whether Indoubt UOWs can exist. If KEYPOINTREQ=YES is specified, and a keypoint is missed, warning message EVE677 is issued during the shutdown. The status will also be changed to stopping if SA z/OS determines that an immediate restart of CICS is unlikely.

KEYPOINTREQ is applicable to CICS TS and pre-CICS TS systems. If KEYPOINTREQ=YES is specified, and warm keypoint data is not available or UOWs indoubt exists, SA z/OS forces the next start type of CICS to be AUTO irrespective of what has been specified in the SIT or by the operator. If a CICS start type of AUTO is always specified, either by the SIT or by the operator, the KEYPOINTREQ parameter is not needed.

At the last shutdown of CICS, if no CICS warm keypoint was taken or if indoubt UOW(s) existed, if KEYPOINTREQ=YES is defined in the automation control file, a message is issued if you have chosen a non-AUTO start type (that is, INITIAL or COLD). You then have the option of specifying override=UOW to force a COLD or INITIAL start. Note however that if the application is being started via a schedule and start type COLD or INITIAL was set (via say INGSET) and if the above mentioned conditions exist, automation will start the application with start type AUTO.

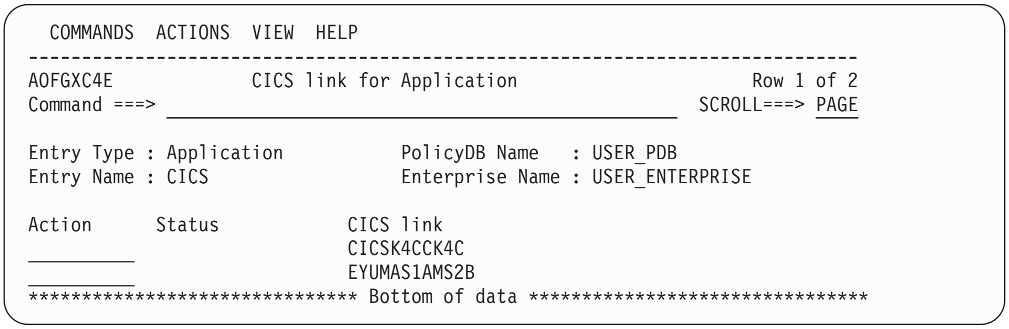
Reload transaction
This specifies a reload transaction to reload the CICS message exit policy on configuration refresh (INGAMS REFRESH). The default value is SARL if nothing is specified. Enter NONE if no reload transaction is desired.

CICSPlex/SM CMAS region
Identifies the CICSPlex/SM CMAS region for shutdown purposes. It must be specified on the application/subsystem that represents the CMAS.

The extended CICS subsystem automation that you define here is also described in the CICSCNTL section in *IBM Tivoli System Automation for z/OS CICS Automation Programmer's Reference and Operator's Guide*.

CICS CONNECTION Policy Item

With this policy item you can select CICS link monitoring definitions required for CICS Automation, as shown in Figure 148.



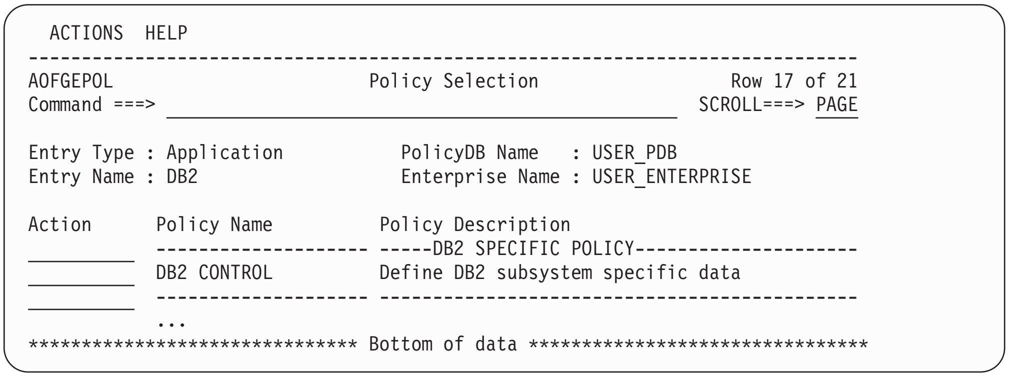
```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E             CICS link for Application                      Row 1 of 2
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : CICS                  Enterprise Name : USER_ENTERPRISE

Action      Status           CICS link
                             CICSK4CCK4C
__________                   EYUMAS1AMS2B
__________
******************************* Bottom of data ********************************
```

Figure 148. CICS Link for Application Panel

This panel allows you to manage the links between the current Application entry and the CICS link entries.

Policy Items for DB2 Subsystems

Figure 149 on page 221 shows the Policy Selection panel for applications that have been created with application type DB2 (see the **Application Type** field in Figure 107 on page 157).



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                  Row 17 of 21
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : DB2                   Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  -------------------- -----DB2 SPECIFIC POLICY---------------------
__________  DB2 CONTROL          Define DB2 subsystem specific data
__________  -------------------- --------------------------------------------
            ...
******************************* Bottom of data ********************************
```

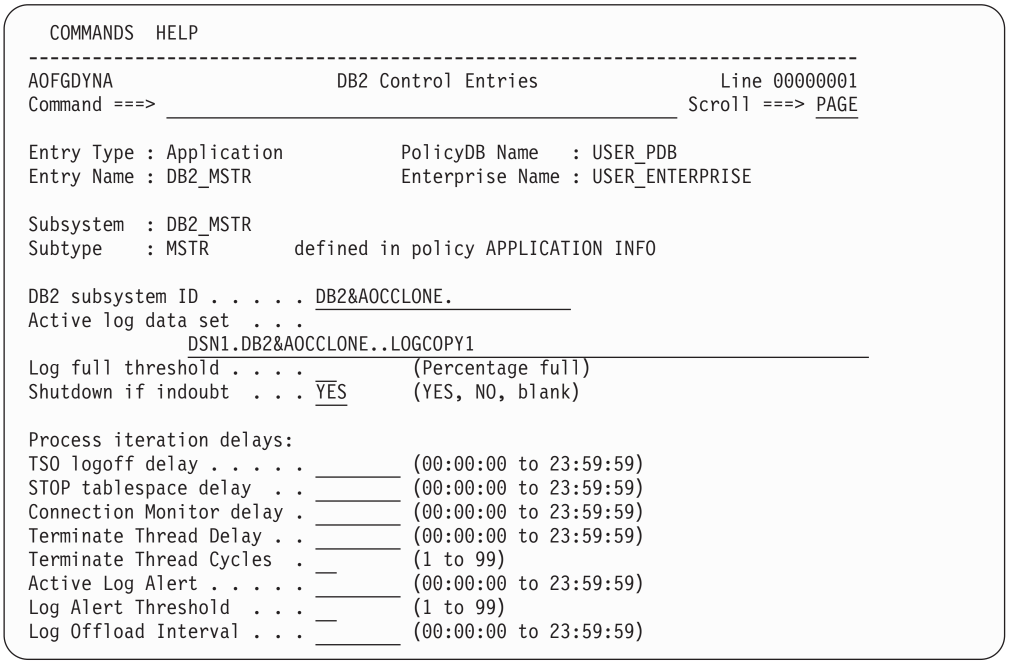
Figure 149. Policy Selection Panel for DB2 Subsystems

The DB2 specific policy items are:

DB2 CONTROL
With this policy item you can specify additional data for a DB2 subsystem. For more information refer to “DB2 CONTROL Policy Item.”

DB2 CONTROL Policy Item

Use this policy item to specify additional data for a DB2 subsystem, shown in Figure 150.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA                     DB2 Control Entries                 Line 00000001
Command ===> ________________________________________________ Scroll ===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : DB2_MSTR              Enterprise Name : USER_ENTERPRISE

Subsystem  : DB2_MSTR
Subtype    : MSTR       defined in policy APPLICATION INFO

DB2 subsystem ID . . . . . DB2&AOCCLONE.
Active log data set  . . .
               DSN1.DB2&AOCCLONE..LOGCOPY1
Log full threshold . . . . __         (Percentage full)
Shutdown if indoubt  . . . YES        (YES, NO, blank)

Process iteration delays:
TSO logoff delay . . . . . ________   (00:00:00 to 23:59:59)
STOP tablespace delay  . . ________   (00:00:00 to 23:59:59)
Connection Monitor delay . ________   (00:00:00 to 23:59:59)
Terminate Thread Delay . . ________   (00:00:00 to 23:59:59)
Terminate Thread Cycles  . __         (1 to 99)
Active Log Alert . . . . . ________   (00:00:00 to 23:59:59)
Log Alert Threshold  . . . __         (1 to 99)
Log Offload Interval . . . ________   (00:00:00 to 23:59:59)
```

Figure 150. DB2 Control Entries Panel

On this panel, either all the fields are inherited from linked classes all together and displayed in a different color, or none of them are.

You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line.

You can override any inherited values by overtyping them. If you override any values for a class, all the inherited values become specific to that particular class or instance (and thus change color to reflect this).

Subtype
This parameter indicates the resource type. It can be entered in the **Subtype** field of the APPLICATION INFO policy item of the application instance or application class. The value shown is either for the application instance or the application class that the application is linked to.

DB2 subsystem ID
This entry should correspond to the DB2 subsystem ID as defined in the system PARMLIB.

You can include system symbols and system automation symbols.

Active Log Dataset
The name of the active log data set to check against before the command to message DSNJ002I is issued. This value is compared with the data set name in the message text. The specified value is implicitly appended with a wildcard character. You can include system symbols and system automation symbols.

Log Full Threshold
How full that the active log data set is allowed to be before the command corresponding to the DSNJ110E message is issued. The default value is 90%.

Shutdown Indoubt
This indicates whether indoubt threads should be checked for before shutting down DB2. Enter YES if the shutdown should proceed without checking for indoubt threads. If you specify NO, using precheck on the NetView shutdown panel still allows checking for indoubt threads.

TSO Logoff Delay
This is the amount of time that the TSO logoff request should allow for before canceling a TSO session. This is used, for example, during thread termination. 1 minute (00:01:00) is the default.

STOP Tablespace Delay
This is the delay between STOP tablespace request cycles. Up to 3 cycles are performed to ensure that the tablespace is shut down. 2 minutes (00:02:00) is the default.

Connection Monitor Delay
This is the delay between connection monitoring cycles. 30 minutes (00:30:00) is the default.

Terminate Thread Delay
This is the amount of time to delay between thread termination cycles. 2 minutes (00:02:00) is the default.

Terminate Thread Cycles
This is the number of cycles to attempt at terminating threads. 6 cycles is the default.

Active Log Alert
This is the amount of time after which the DSNJ111E alert is reset. The default value is 5 minutes (00:05:00).

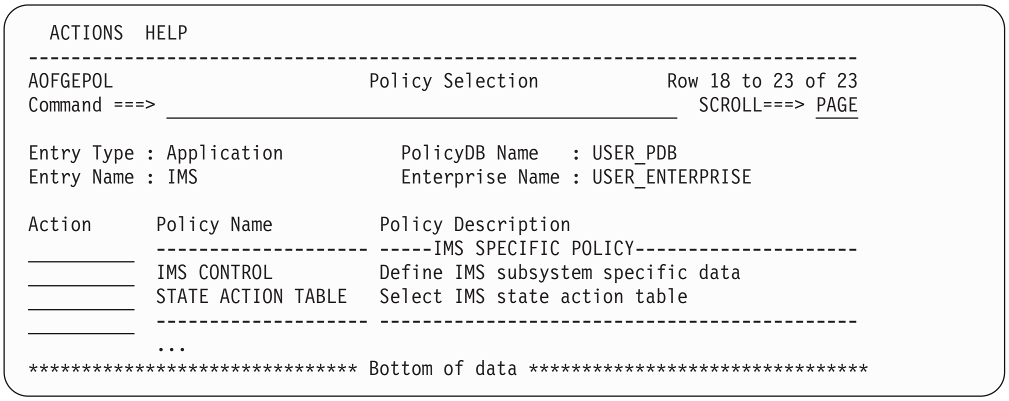
Log Alert Threshold
This is the number of DSNJ111E alerts to allow for before issuing any associated commands. 5 alerts is the default.

Log Offload Interval
This is the amount of time to allow between DSNJ115I messages before issuing an alert plus any associated commands for the DSNJ115I message entry. 30 minutes is the default.

Policy Items for IMS Subsystems

Figure 151 shows the Policy Selection panel for applications that have been created with application type IMS (see the **Application Type** entry field in Figure 107 on page 157). This Policy Selection panel contains additional policy items for an IMS subsystem.

Some of these policy items are only applicable to an IMS subsystem defined as an instance. If a subsystem CLASS of type IMS is selected, only the additional policy items IMS CONTROL and STATE ACTION TABLE are available.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection            Row 18 to 23 of 23
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : IMS                  Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
_________   -------------------- -----IMS SPECIFIC POLICY---------------------
_________   IMS CONTROL          Define IMS subsystem specific data
_________   STATE ACTION TABLE   Select IMS state action table
_________   -------------------- ---------------------------------------------
            ...
****************************** Bottom of data ********************************
```

Figure 151. Policy Selection Panel for IMS Subsystems

The IMS specific policy items are:

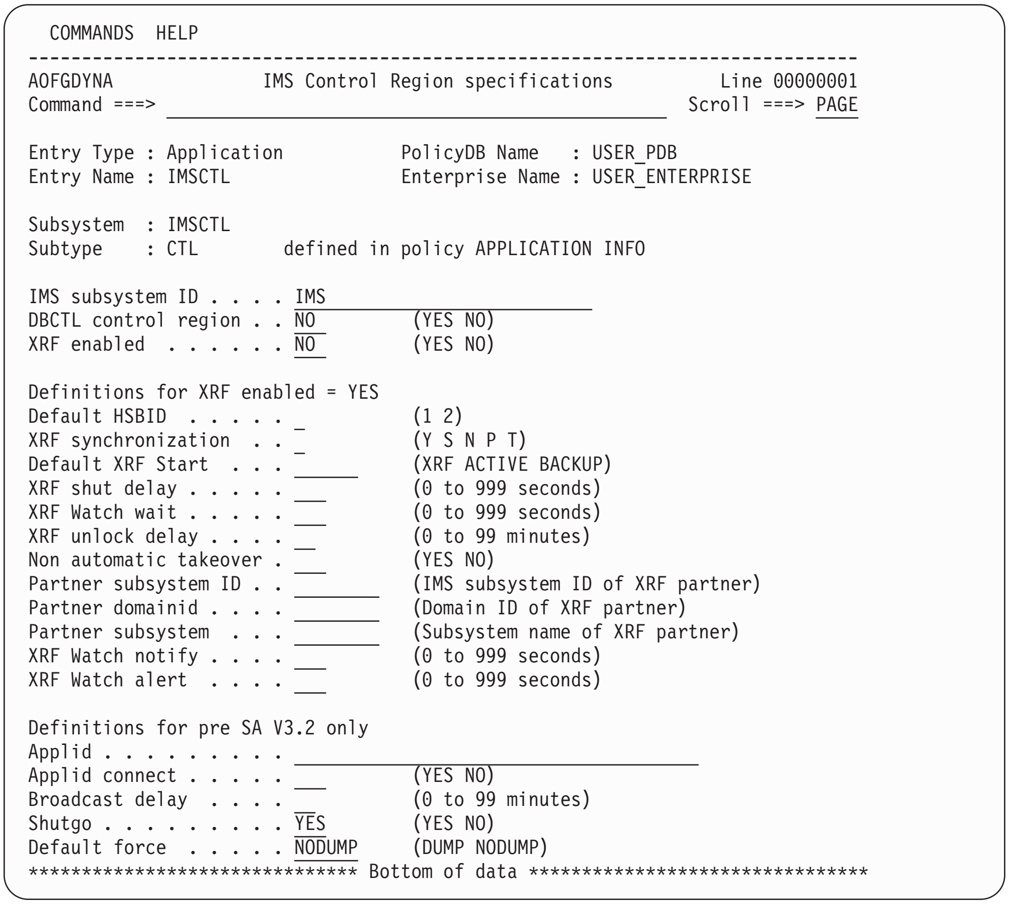
IMS CONTROL
Use this policy item to define additional IMS information for the IMS CONTROL REGION subsystem. This is described in *IBM Tivoli System Automation for z/OS IMS Automation Programmer's Reference and Operator's Guide* in section IMSCNTL. This policy item is only defined for an IMS control region. For more information refer to “IMS CONTROL Policy Item.”

STATE ACTION TABLE
If you select this policy item, SA z/OS will display a list of state/action tables which previously have been defined as described in “IMS State/Action Tables Policy Object” on page 261. From this list you can link one state/action table to the current IMS subsystem or change or remove an existing link.

Note that this policy item is only listed for systems that are running an earlier release of SA z/OS.

IMS CONTROL Policy Item

If you select the IMS CONTROL policy item from the Policy Selection panel for IMS subsystems, and you have specified the subtype CTL, Figure 152 on page 224 is displayed where you describe the IMS/DC control region.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA              IMS Control Region specifications          Line 00000001
Command ===> _______________________________________________  Scroll ===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : IMSCTL                Enterprise Name : USER_ENTERPRISE

Subsystem  : IMSCTL
Subtype    : CTL       defined in policy APPLICATION INFO

IMS subsystem ID . . . . IMS_____________________
DBCTL control region . . NO_         (YES NO)
XRF enabled  . . . . . . NO_         (YES NO)

Definitions for XRF enabled = YES
Default HSBID  . . . . . _           (1 2)
XRF synchronization  . . _           (Y S N P T)
Default XRF Start  . . . ______      (XRF ACTIVE BACKUP)
XRF shut delay . . . . . ___         (0 to 999 seconds)
XRF Watch wait . . . . . ___         (0 to 999 seconds)
XRF unlock delay . . . . __          (0 to 99 minutes)
Non automatic takeover . ___         (YES NO)
Partner subsystem ID . . ________    (IMS subsystem ID of XRF partner)
Partner domainid . . . . ________    (Domain ID of XRF partner)
Partner subsystem  . . . ________    (Subsystem name of XRF partner)
XRF Watch notify . . . . ___         (0 to 999 seconds)
XRF Watch alert  . . . . ___         (0 to 999 seconds)

Definitions for pre SA V3.2 only
Applid . . . . . . . . . ______________________________________
Applid connect . . . . . ___         (YES NO)
Broadcast delay  . . . . __          (0 to 99 minutes)
Shutgo . . . . . . . . . YES         (YES NO)
Default force  . . . . . NODUMP      (DUMP NODUMP)
******************************* Bottom of data ********************************
```

Figure 152. IMS Control Region Specifications Panel for subtype CTL

The fields that you enter or update have the following meaning:

Subtype
This parameter shows the subsystem type. It is the only required parameter and can be entered on the APPLICATION INFO policy item of the application, or the application class in the **Subtype** field. The value shown is either the one specified for the application or the application class that the application is linked to.

IMS Subsystem ID
Specifies the subsystem ID (IMSID) specified in the IMS control region parameters or SYSGEN.

DBCTL control region
Specify YES to indicate that this is an IMS DBCTL control region, or NO to indicate that it is an IMS DB/DC control region.

Note: If you specify YES in the **DBCTL control region** field, you must specify NO in the **XRF enabled** field.

XRF enabled
This parameter specifies whether the control region will execute as an XRF complex.

Note:

Specifying NO in the **XRF enabled** field causes all the XRF-specific fields to be automatically forced to blank.

The following fields are *only* valid if you specify YES in the **XRF enabled** field.

Default HSBID
The DEFLTHSB specification describes the preferred HSBID to be used when restarting an IMS system in situations where it does not matter which HSBID is used (in terms of IMS recovery).

XRF Synchronization
This parameter is used to control the type and degree of region synchronization between the active and the alternate IMS subsystems. This parameter is effective when the parent IMS control region is the alternate. The parameter is not used while the parent is the active IMS.

Y

When the partner region is stopped or started on the active IMS subsystem this region is also stopped or started on the alternate IMS subsystem to mirror the active IMS subsystem.

N This region will not participate in any type of synchronization.

P This region will not participate in any type of synchronization; however it will be started when the alternate IMS is started.

S When the partner region is started on the active IMS subsystem, this event is recorded. If a takeover occurs, the region will be started on the alternate after takeover processing is complete.

T This region will be started after the IMS takeover.

Default XRF Start
This parameter is used to define the default XRF start type for the control region startup panel to be used for a control region startup.

XRF shut delay
During shutdown of an XRF IMS complex with an alternate that is executing, IMS automation will attempt the shutdown of the alternate prior to the active. For an "Immediate" type of shutdown, shut delay is the time in seconds to wait after initiating alternate shutdown before continuing with the active shutdown process.

XRF Watch wait
During XRF takeover processing, this time is the number of seconds between each invocation of takeover watch or monitor processing.

XRF unlock delay
During an XRF takeover, once the UNLOCK command has been issued by IMS automation surveillance processing this parameter specifies the number of minutes to suspend the surveillance CLIST. After the suspension, the CLIST is invoked again to check XRF takeover processing.

Non automatic takeover
If the XRF complex is defined so that a takeover will not automatically occur (specified in the DFSHSB*xx* member of the IMS PROCLIB), a message will be issued to the MVS system console so the operator can respond to allow the takeover to occur or to stop the process. If the parameter specifies YES, IMS automation will respond with the affirmative reply to this message.

Partner subsystem ID
Use this parameter to specify the subsystem ID for the partner IMS control region. This control region may be specified in either the local or a remote domain.

Partner domainid
Use this parameter to specify the NetView domain ID for the partner IMS control region. This may be the local or a remote domain.

Partner subsystem
Use this parameter to specify the SA z/OS subsystem name of the partner IMS control region.

XRF Watch notify
Use this parameter to tailor XRF takeover surveillance processing. It is specified in seconds.

If surveillance processing does not detect a change in the XRF status for the monitored IMS subsystem before the time specified in this field, additional IMS automation warning messages will be issued. These messages describe possible problems with the takeover. Once issued, the clock starts again and the messages are not issued until the time spent in surveillance once again exceeds the time specified in watch notify.

XRF Watch alert
This parameter is used to tailor XRF takeover surveillance processing. It is specified in seconds.

Once surveillance processing exceeds the time specified by this parameter without a change in the XRF status, additional messages are issued indicating that operator intervention may be required to complete the takeover. Once issued, the messages will be repeatedly issued with the interval that is specified in the **XRF Watch wait** field until an XRF status change is detected.

The following definitions

Applid
This identifies the receiver/sender name when the program-to-program interface is initiated and used in an IMS DB/DC control region. Use one of the following:

Applid
Use this keyword if this subsystem is defined as non-XRF.

(*generic-applid***,***specific-applid***)**
If this subsystem is defined as XRF-capable, use *generic-applid* as the name given to both the active and alternate subsystem, and *specific-applid* as the specific name for this subsystem.

Applid connect
During IMS startup, if this is specified as YES, the IMS DB/DC control region will not be started unless the APPLID is connectable.

Regardless of what is specified in this field, an XRF alternate subsystem will not be started if the APPLID is not connectable.

Broadcast delay
If the BROADCAST or ABORT SHUTDOWN option is selected on one of the shutdown panels, this parameter defines the time in minutes between the execution of the /BRO command and the start of the shutdown (SHUTSYS) process of the IMS DB/DC control region.

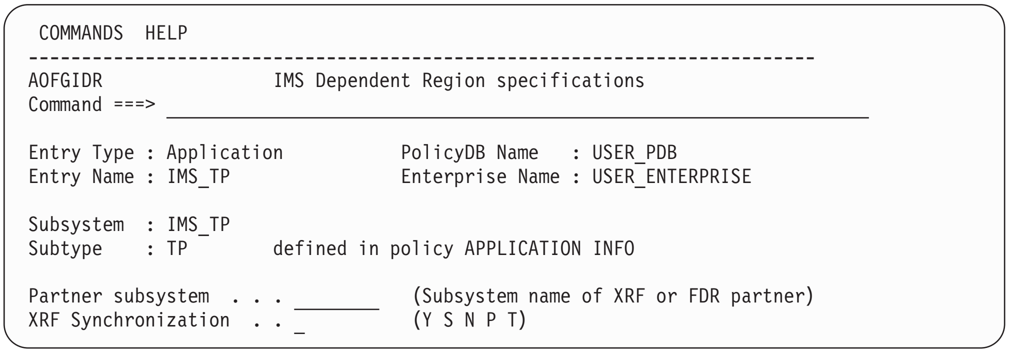
Shutgo

Specifying YES will allow shutdown of a control region to progress from a normal shutdown, then to an immediate shutdown, then to a forced shutdown (as needed) without any operator action.

Default force

This parameter is used during an escalation of shutdown from type IMMED to type FORCE. It retrieves the default shut options for a type FORCE shutdown. This parameter specifies the dump option associated with a type FORCE shutdown.

If the specified subtype is not CTL, the panel shown in Figure 153 is displayed.



```
 COMMANDS  HELP
------------------------------------------------------------------------------
AOFGIDR                IMS Dependent Region specifications
Command ===> __________________________________________________________________

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : IMS_TP                Enterprise Name : USER_ENTERPRISE

Subsystem  : IMS_TP
Subtype    : TP        defined in policy APPLICATION INFO

Partner subsystem  . . . ________   (Subsystem name of XRF or FDR partner)
XRF Synchronization  . . _          (Y S N P T)
```

Figure 153. IMS Dependent Region Specifications Panel

Partner subsystem

This parameter is used for XRF enabled systems only. For IMS dependent regions this is the SA z/OS subsystem name of the corresponding XRF partner dependent regions.

XRF synchronization

This parameter is used for XRF systems only and controls the type and degree of region synchronization between the active and the alternate IMS subsystems. This parameter is effective when the parent IMS control region is the alternate. While the parent is the active IMS, the parameter is not used.

Y When the partner region is stopped or started on the active IMS subsystem this region is also stopped or started on the alternate IMS subsystem to mirror the active.

N This region will not participate in any type of synchronization.

P This region will not participate in any type of synchronization, however, it will be started when the alternate IMS is started.

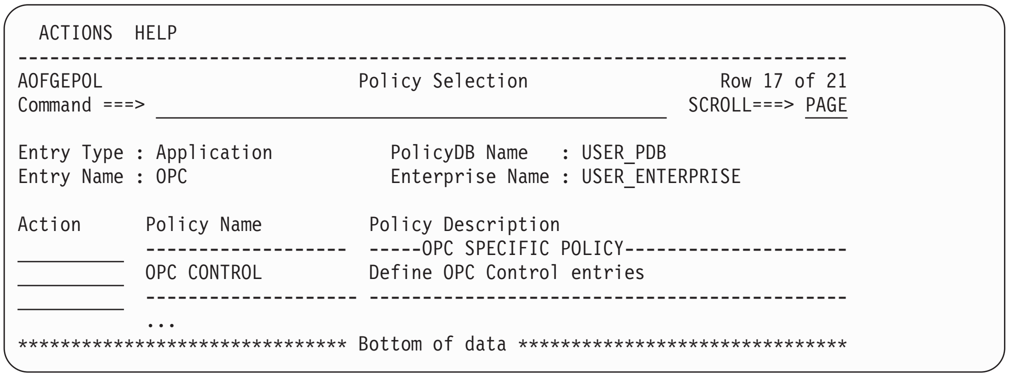
S When the partner region is started on the active IMS subsystem, this is recorded. The region will be started on the alternate when a takeover occurs after takeover processing is complete.

T This region will be started after IMS takeover.

Policy Items for OPC Subsystems

Figure 154 on page 228 shows the Policy Selection panel for applications that have been created with application type OPC (see the **Application Type** entry field in Figure 107 on page 157). This Policy Selection panel contains additional policy items for an OPC subsystem.

These policy items are only applicable to an OPC subsystem defined as an instance. If a subsystem CLASS of type OPC is selected, no additional policy items are available.



```
  ACTIONS  HELP
--------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                   Row 17 of 21
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : OPC                   Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  -------------------  -----OPC SPECIFIC POLICY---------------------
__________  OPC CONTROL          Define OPC Control entries
__________  -------------------- ---------------------------------------------
            ...
****************************** Bottom of data *******************************
```

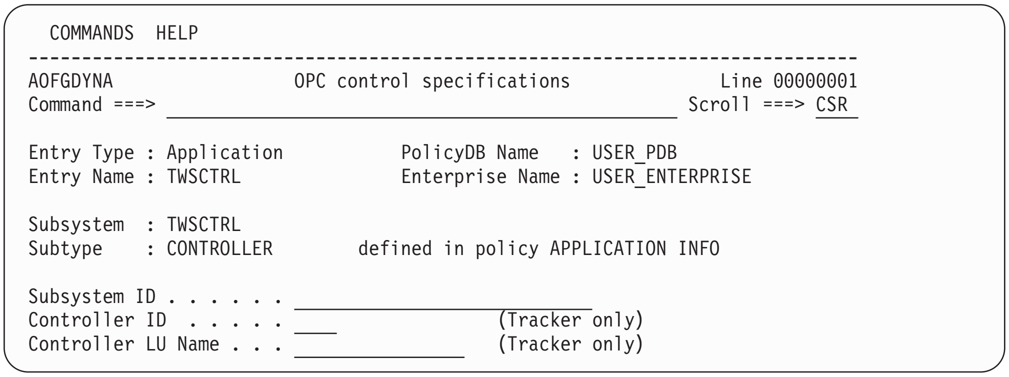
Figure 154. Policy Selection Panel for OPC Subsystems

The TWS specific policy item is:

OPC CONTROL
: Use this policy item to specify TWS extensions to subsystem definitions. For more information, see “OPC CONTROL Policy Item.”

OPC CONTROL Policy Item

If you select the OPC CONTROL policy item on the Policy Selection panel for OPC subsystems, SA z/OS displays the panel in Figure 155 where you can describe additional information for OPC subsystems.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA                OPC control specifications              Line 00000001
Command ===> ________________________________________________ Scroll ===> CSR

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : TWSCTRL               Enterprise Name : USER_ENTERPRISE

Subsystem  : TWSCTRL
Subtype    : CONTROLLER        defined in policy APPLICATION INFO

Subsystem ID . . . . . . ____________________________
Controller ID  . . . . . ____                  (Tracker only)
Controller LU Name . . . ________________      (Tracker only)
```

Figure 155. OPC control specifications Panel

Subtype
: This parameter shows the subtype that was specified in the APPLICATION INFO policy item for the application itself or application class that this application is linked to.

Fill in the following fields:

Subsystem ID
: Enter the OPC subsystem ID.

Controller ID
: Enter the OPC subsystem ID of the controller that is controlling this tracker. This field is only used for tracker definitions. It allows for the specification of a remote controller. If the controller is not in the same sysplex as the tracker, specify this field.

Note: Remote controllers with the hot standby function are not supported.

Controller LU Name
Enter the Logical Unit name of the controller that is controlling this tracker. This field is only used for tracker definitions. It allows for the specification of a remote controller. If the controller is not in the same sysplex as the tracker, specify this field.

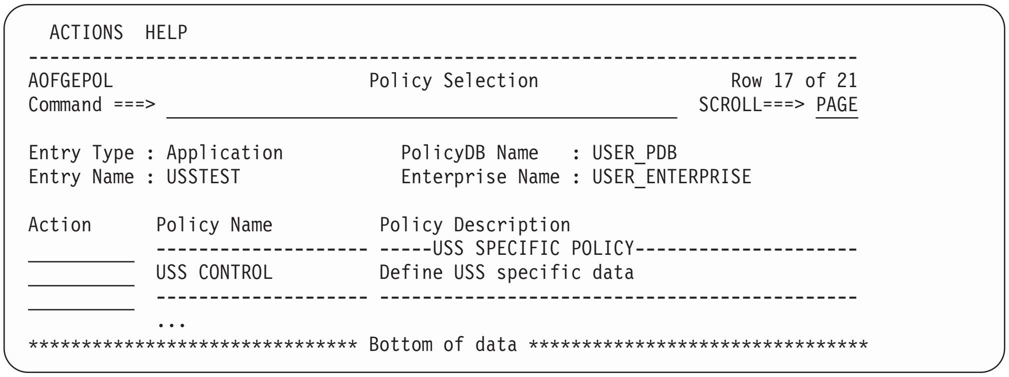
You do not need to specify the **Controller ID** and **Controller LU Name** fields if the controller is in a remote sysplex or system that can be controlled by SA z/OS with the TARGET= parameter of the commands.

Policy Items for USS Subsystems

Figure 156 shows the Policy Selection panel for applications that have been created with application type USS (see "Application Type" on page 159).

USS Applications must be defined with a HASPARENT relationship to JES.

By using the USS CONTROL policy item you can define UNIX System Services specific data.

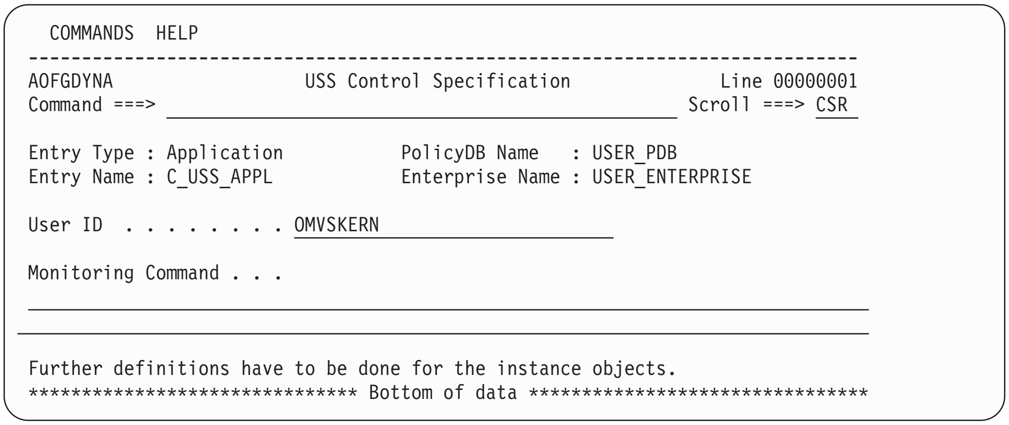


```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                  Row 17 of 21
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Application           PolicyDB Name   : USER_PDB
Entry Name : USSTEST               Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  -------------------- -----USS SPECIFIC POLICY---------------------
__________  USS CONTROL          Define USS specific data
__________  -------------------- ---------------------------------------------
            ...
****************************** Bottom of data ********************************
```

Figure 156. Policy Selection Panel for USS Subsystems

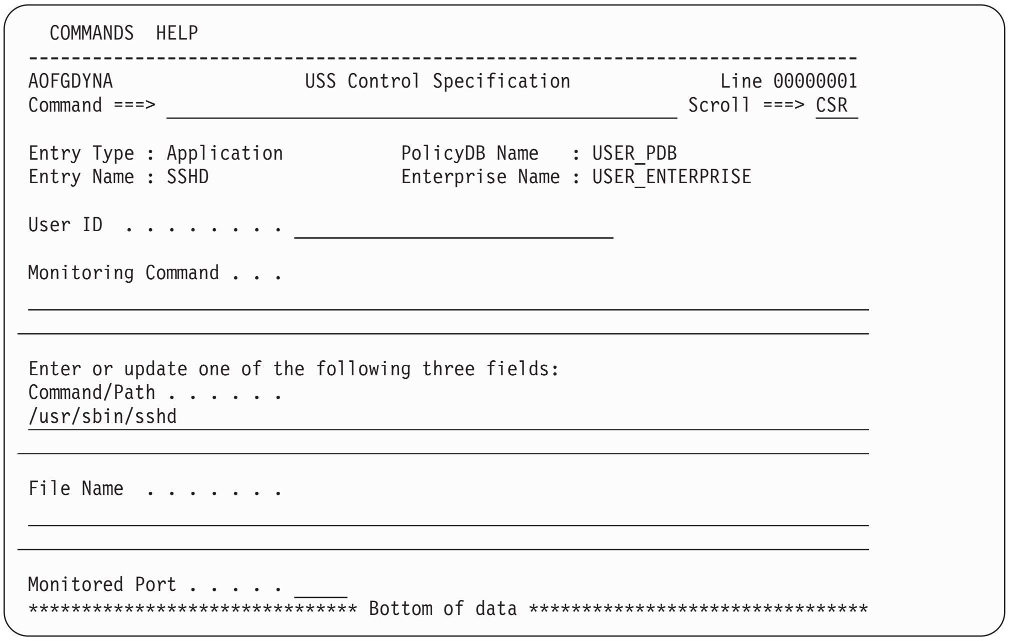
USS CONTROL Policy Item

If you select the USS CONTROL policy item on the Policy Selection panel for USS subsystems, SA z/OS displays the panel shown in Figure 157 on page 230 and Figure 158 on page 230 where you can enter additional information for UNIX subsystems.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA                 USS Control Specification             Line 00000001
Command ===>                                                  Scroll ===> CSR

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : C_USS_APPL           Enterprise Name : USER_ENTERPRISE

User ID  . . . . . . . . OMVSKERN

Monitoring Command . . .



Further definitions have to be done for the instance objects.
******************************* Bottom of data ********************************
```

Figure 157. USS Control Specification Panel for Application Classes



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNA                 USS Control Specification             Line 00000001
Command ===>                                                  Scroll ===> CSR

Entry Type : Application          PolicyDB Name   : USER_PDB
Entry Name : SSHD                 Enterprise Name : USER_ENTERPRISE

User ID  . . . . . . . .

Monitoring Command . . .



Enter or update one of the following three fields:
Command/Path . . . . . .
/usr/sbin/sshd


File Name  . . . . . . .



Monitored Port . . . . .
******************************* Bottom of data ********************************
```

Figure 158. USS Control Specification Panel for Application Instances

If either of the values for the **User ID** or **Monitoring Command** has been inherited from linked classes, they are displayed in a different color. You can find out which class a value has been inherited from by typing the command `classname` (or its abbreviation `cn`) on the command line, moving the cursor to the field of interest and pressing Enter. You can override any inherited values by overtyping them. If you override either of the values that have been inherited, they are individually applied to the current application and change color to turquoise to indicate this.

Note: For application classes you can specify only the User ID and Monitoring Command.

For application instances, you can specify the User ID and Monitoring Command and *one* of the other three fields.

User ID

Enter the UNIX user ID that this resource belongs to.

Monitoring Command

Enter the UNIX System Services monitoring command that is used to get the status of the resource. This command must be a valid program in the HFS. If the field is left blank, the SA z/OS supplied standard monitoring routine is used. The return codes must be:

0 The resource is available.

4 The resource is starting.

8 The resource is unavailable.

12 An error occurred.

If the first character is a ’/’, the specified command is considered to be an absolute path. Otherwise the monitoring routine must be in the same directory as the SA z/OS routines.

Command/Path

Enter the name of the command that is running without its arguments, as shown by the UNIX **ps** command with the format specification **comm**.

File Name

Enter the path of a file in the HFS. If the file exists and the user ID that you specified is the owner of the file the resource is available, otherwise the resource is not available.

Monitored Port

Enter the TCP port number that this resource is expected to listen to on the local host.

Defining a VTAM Application to SA z/OS

VTAM applications need to have Application nodes activated for the application to operate correctly. This is normally not a problem if an application is to run on a single system. However, if the application is to be switched from one system to another (via a move group or server group), the application node definition must be deleted from the system that the application is moving from. If this is not done, users may not be able to log on to the application because there is a definition for the application node that is not active, that is, the application has not opened the node ACB.

To alleviate this problem, the application node must be deleted from the old system and created on the new system. Unfortunately, the only way to delete a node in VTAM is to deactivate its major node, that is, the member that it is defined in.

Newer releases of VTAM have introduced the concept of *Model node definitions*. In this case a major node is created with the type of node and a name that includes wildcards. Whenever a node of the type is accessed, VTAM will use the name requested to match the models. It will then dynamically create a node based on the model definitions with the name requested. When the node is no longer required it will delete it. What this means for application nodes is that a model definition can be defined once on each VTAM in the network that the application might be run on. Then when the application is started and opens its ACB, VTAM will dynamically create the node for it. Likewise when moving the application, upon closing the ACB, VTAM will delete the node and another VTAM on another system will dynamically create the node.

VTAM applications may require recovery commands to be issued if VTAM is restarted, or the VTAM application node is reactivated. These commands differ from subsystem to subsystem and can be specified in the Messages/User Data policy items as described in the following section.

The INGVTAM command provides a method of activating the Major Nodes for an application before the application is started, and deactivating the Major Nodes after it is down. To enable the function you must code the INGVTAM command in the prestart, ACORESTART, post-shut policies. In addition, if VTAM should ever be restarted whilst the applications are running, the major nodes must be reactivated. This can be accomplished by coding the INGVTAM command in the UP messages/user data policy for the VTAM subsystem.

Registering VTAM Application Subsystems with SA z/OS Recovery

To enable VTAM application recovery to take place, the subsystems must be registered with the SA z/OS recovery code. This is achieved by using the INGVTAM command that is described in *IBM Tivoli System Automation for z/OS Programmer's Reference*. The following application policy items must be customized:

1. PRE-START policy

   The PRE-START policy must have at least a NORMal start item with the INGVTAM command to activate a list of major nodes. The following command can be used as an example:

   ```
   INGVTAM &SUBSAPPL REQ=ACTIVATE MAJNODE=(majnode1,majnode2,...)
   ```

   Where `majnode`*n* are VTAM application major nodes. Each major node will be varied active to VTAM when the subsystem prestart commands are issued. Note, it is expected that only one of the major nodes will contain the minor node that the VTAM application subsystem will use.

2. POST-SHUTDOWN policy

   The POST-SHUTDOWN policy is used to deregister the subsystem with SA z/OS VTAM application recovery. Use the INGVTAM REQ=DEACTIVATE command in the policy. For example:

   ```
   INGVTAM &SUBSAPPL REQ=DEACTIVATE
   ```

   The DEACTIVATE request issues a vary net inactive for each major node registered by the REQ=ACTIVATE. The vary is not done if the major node is shared by other subsystems that have also registered the major node. When the last subsystem registered issues an INGVTAM REQ=DEACTIVATE, the major node will be varied inactive. The only exception to this is when the major node contains model resources with wildcards in the node definition. In this case the major node is never inactivated.

3. ACORESTART Messages/User Data policy

   The ACORESTART message policy must have the same definition as the PRE-START policy. This policy item is used to reregister the subsystem with SA z/OS VTAM application recovery.

4. VTAMUP Messages/User Data policy

   Enter commands that are issued when the VTAM subsystem is restarted. Typically these commands reopen the VTAM ACB that the subsystem uses to communicate with VTAM.

5. Relationships policy

   Optionally enter a relationship for the subsystem to ensure that the prestart commands are only issued when VTAM is up. The required relationship is:

```
PrepAvailable(WhenAvailable),Passive,Weak -> VTAM/APL/=
```

Where `VTAM` is the name of the VTAM subsystem and that is the supporting resource. `Passive` forces the relationship to wait until VTAM is UP. `Weak` specifies that only the supporting resource status is checked.

In addition the UP message for VTAM should have the following command:

```
INGVTAM REQ=ACTIVATE
```

When INGVTAM is executed with REQ=ACTIVATE and no positional subsystem, it finds all the subsystems that had previously registered via INGVTAM and issues Vary NET ACT commands for their major nodes. After this has been done, it will execute any policy command(s) that is/are specified to USER MESSAGE VTAMUP for the subsystems.

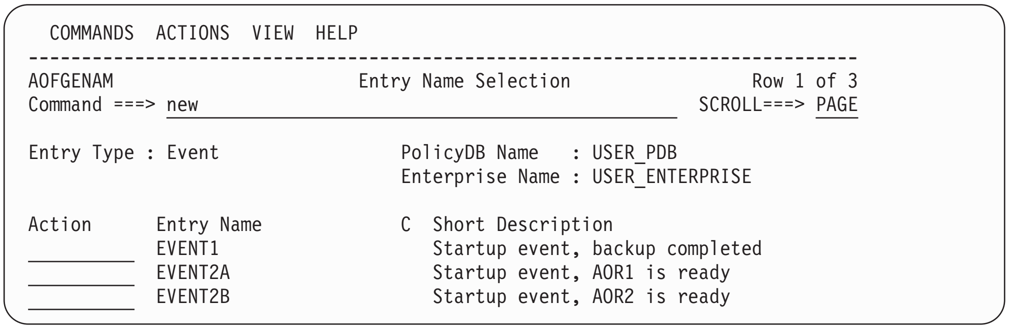
Events Policy Object

Events basically are used for external dependencies where they represent a process that the automation cannot handle. For example, an operator needs to manually edit a file before the application can be started. SA z/OS remembers which events are SET (have occurred) and can be told when to forget them (by defining an UNSET condition).

Creating a New Event

To define a new event, enter `7` or `EVT` for events in the panel shown in Figure 2 on page 7.

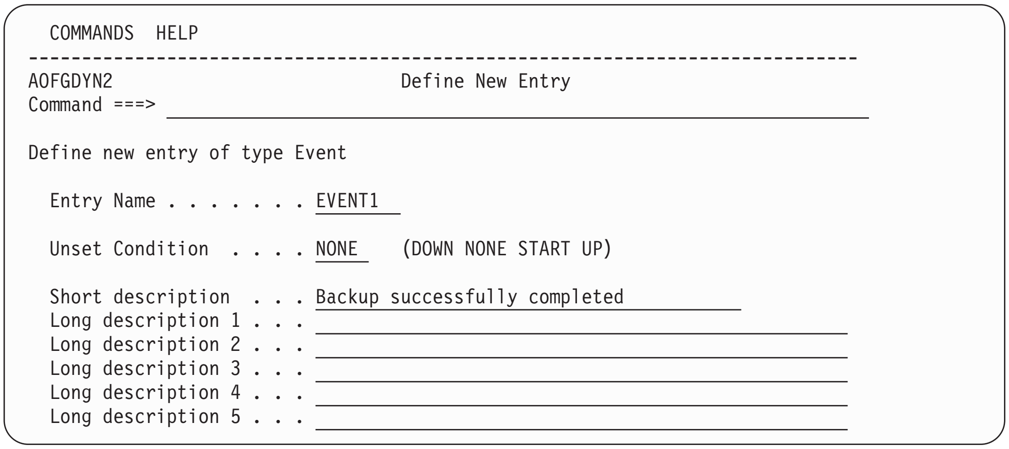
This displays a panel similar to the one shown in Figure 159. This panel may show a list of already existing events. Enter the command **new** on the command line.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGENAM                       Entry Name Selection                 Row 1 of 3
Command ===> new                                              SCROLL===> PAGE

Entry Type : Event                 PolicyDB Name   : USER_PDB
                                   Enterprise Name : USER_ENTERPRISE

Action      Entry Name             C  Short Description
__________  EVENT1                    Startup event, backup completed
__________  EVENT2A                   Startup event, AOR1 is ready
__________  EVENT2B                   Startup event, AOR2 is ready
```

Figure 159. Entry Name Selection Panel for Events

A panel similar to the one shown in Figure 160 on page 234 is displayed. Specify the event name, the UNSET condition, and a description.



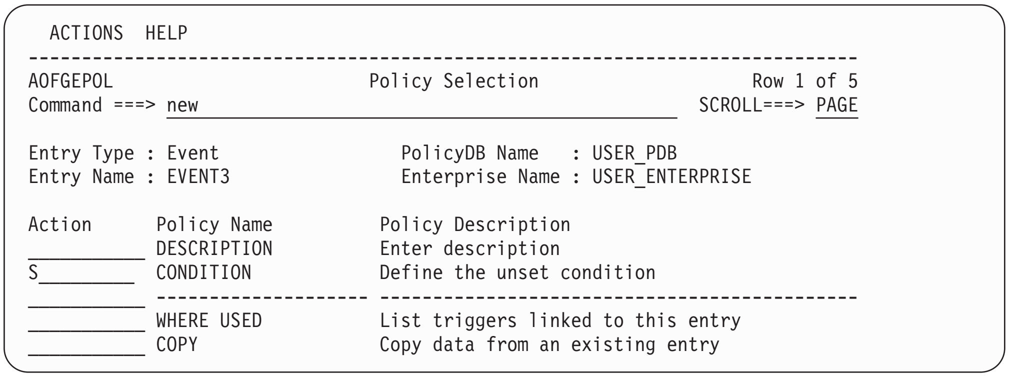
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN2                           Define New Entry
Command ===> ___________________________________________________________________

Define new entry of type Event

  Entry Name . . . . . . . EVENT1__

  Unset Condition  . . . . NONE_    (DOWN NONE START UP)

  Short description  . . . Backup successfully completed__________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 160. Define New Event Panel

The unset condition is preset to NONE. You can change this input field if desired. For more information on the unset condition, see "CONDITION Policy Item."

Policy Items for Events

Figure 161 shows the policy items that are displayed when an event is selected from the Entry Name Selection panel for events.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                   Row 1 of 5
Command ===> new_______________________________________________  SCROLL===> PAGE

Entry Type : Event                 PolicyDB Name   : USER_PDB
Entry Name : EVENT3                Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
___________ DESCRIPTION          Enter description
S_________  CONDITION            Define the unset condition
___________ -------------------- ---------------------------------------------
___________ WHERE USED           List triggers linked to this entry
___________ COPY                 Copy data from an existing entry
```

Figure 161. Policy Selection Panel for Events

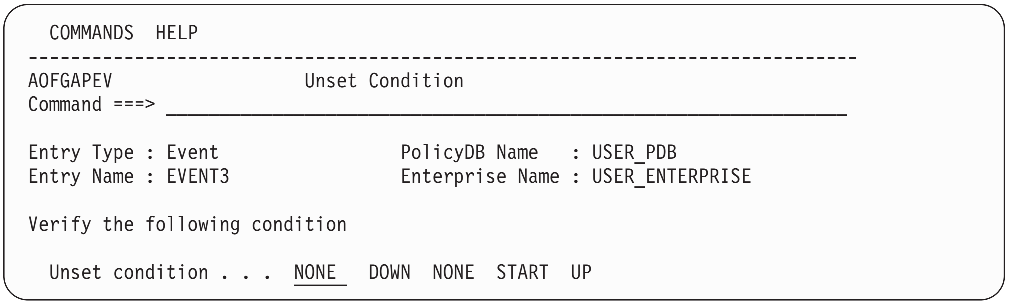
The unique policy item for Events is:

CONDITION
With this policy item you can specify when SA z/OS should unset (forget) a previously set event.

Note:
On the WHERE USED panel for Events, only action "remove" is available.

CONDITION Policy Item

Selecting the CONDITION policy item will bring up a panel as shown in Figure 162 on page 235 where you can define the *unset condition* of the current event.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGAPEV                 Unset Condition
Command ===> ________________________________________________________________

Entry Type : Event                PolicyDB Name   : USER_PDB
Entry Name : EVENT3               Enterprise Name : USER_ENTERPRISE

Verify the following condition

  Unset condition . . .  NONE   DOWN  NONE  START  UP
```

Figure 162. Unset Condition Panel for Events

The unset condition determines when an event for an application is reset to *not occurred* (that is, unset). You can enter the following values:

NONE
Once set, the event will never be unset.

START
Once set, the event will be unset if the application is already in the respective state or if the observed status of the application turns into STARTING.

UP
Once set, the event will be unset if the application is already in the respective state or if the observed status of the application turns into one of the following:
- AVAILABLE
- DEGRADED

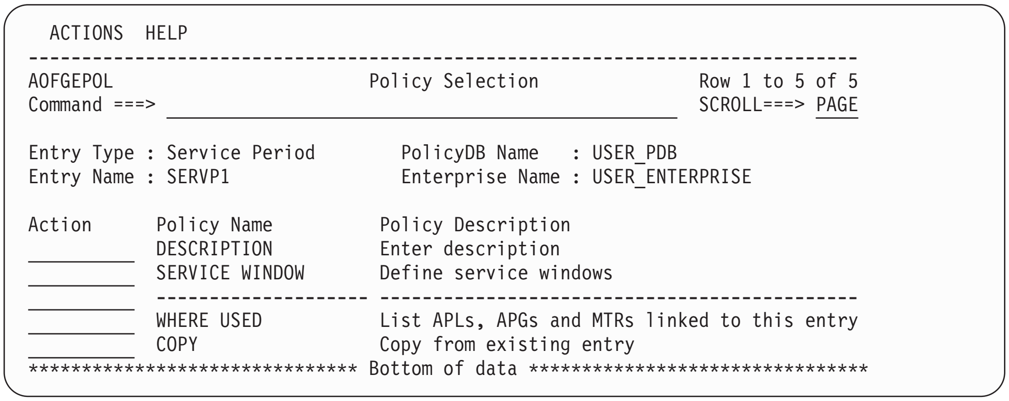
DOWN
Once set, the event will be unset if the application is already in the respective state or if the observed status of the application turns into one of the following:
- SOFTDOWN
- HARDDOWN
- SYSGONE
- STANDBY

Service Periods Policy Object

Service periods define the availability of resources. A service period is a set of time intervals (service windows) during which a resource should be active or inactive.

Policy Items for Service Periods

Figure 163 on page 236 shows the policy items that are displayed when an existing Service Periods entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection              Row 1 to 5 of 5
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Service Period        PolicyDB Name   : USER_PDB
Entry Name : SERVP1                Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  SERVICE WINDOW       Define service windows
__________  -------------------- ---------------------------------------------
__________  WHERE USED           List APLs, APGs and MTRs linked to this entry
__________  COPY                 Copy from existing entry
******************************* Bottom of data ********************************
```

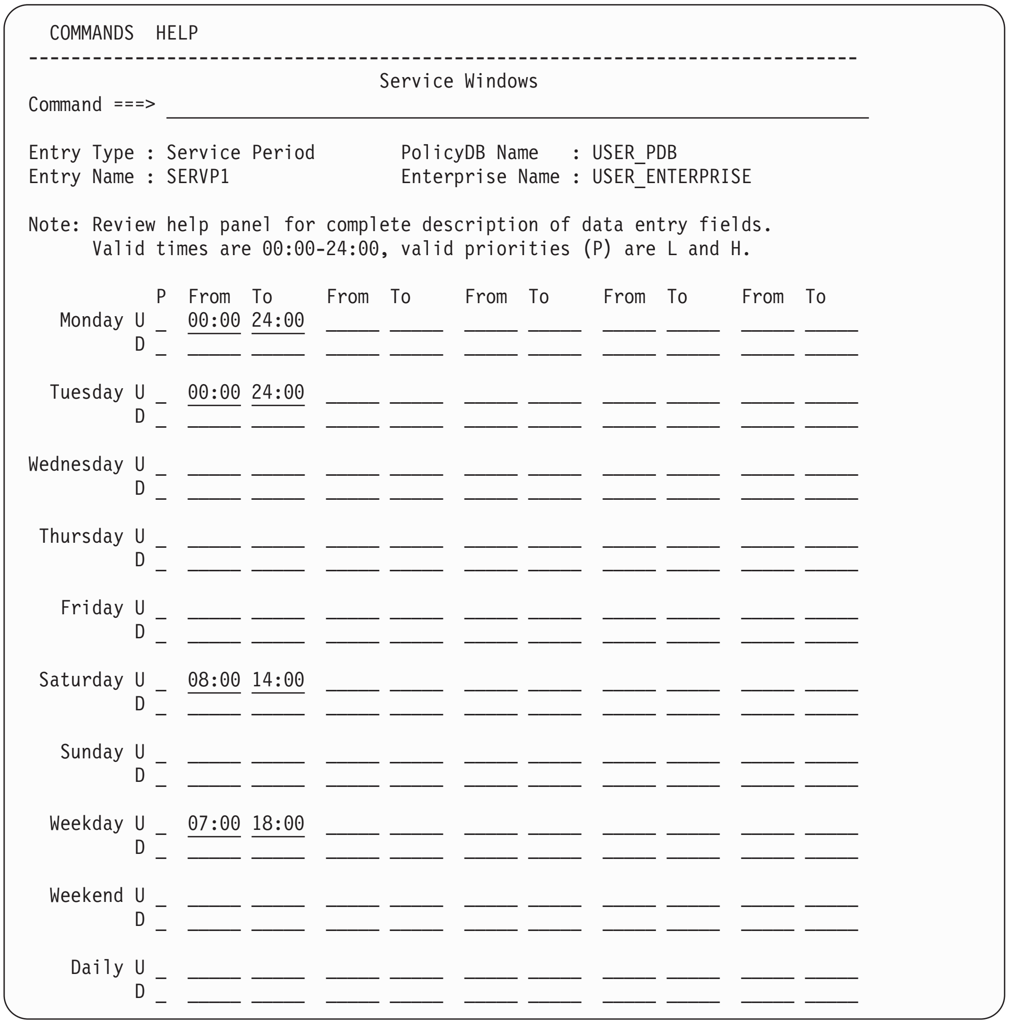
Figure 163. Policy Selection Panel for Service Periods

The unique policy item for Service Periods is:

SERVICE WINDOW
With this policy item you define the time frame (or frames) when an application, application group or monitor resource should be available or unavailable. You need to link a service period with a defined service window to Applications, Application Groups, or Monitor Resources.

SERVICE WINDOW Policy Item

Selecting the SERVICE WINDOW policy item will bring up a panel as shown in Figure 164 on page 237 where you can define the time duration of the service period.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
                                 Service Windows
Command ===> __________________________________________________________________

Entry Type : Service Period        PolicyDB Name   : USER_PDB
Entry Name : SERVP1                Enterprise Name : USER_ENTERPRISE

Note: Review help panel for complete description of data entry fields.
      Valid times are 00:00-24:00, valid priorities (P) are L and H.

           P  From  To     From  To     From  To     From  To     From  To
   Monday U _ 00:00 24:00  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

  Tuesday U _ 00:00 24:00  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

Wednesday U _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

 Thursday U _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

   Friday U _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

 Saturday U _ 08:00 14:00  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

   Sunday U _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

  Weekday U _ 07:00 18:00  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

  Weekend U _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

    Daily U _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
          D _ _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
```

Figure 164. Service Windows Panel for Service Periods

A service period is a schedule of times (referred to as service windows) when a resource is expected to be active or inactive. Times that you enter into a line preceded by letter *U* will be used as From/To times for the periods when the resource should be available (UP). A line preceded by letter *D* will contain From/To times for DOWN periods. Start and Stop times are specified in 24 hour notation, 00:00-24:00, and in any order. For example, to keep an application continuously available from Monday through Tuesday around the clock, specify the times as shown:

```
          P  From  To     From  To     From  To     From  To     From  To
 Monday U _  00:00 24:00  _____ _____  _____ _____  _____ _____  _____ _____
        D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____

Tuesday U _  00:00 24:00  _____ _____  _____ _____  _____ _____  _____ _____
        D _  _____ _____  _____ _____  _____ _____  _____ _____  _____ _____
```

The schedule is defined by specifying one or more (up to five) service windows for:

- **each day of the week:** Monday through Sunday
- **weekday days:** the weekday schedule (Monday through Friday)
- **weekend days:** the weekend schedule (Saturday and Sunday)
- **Daily:** all seven days; the daily schedule

For the time between the specified intervals, the automation manager decides what to do with the resources.

For each UP and DOWN schedule, in the **P** column, you may specify a priority:

L for low priority; Orders resulting from a service period with low priority can be overruled by an operator's INGREQ command. This is the default.

H for high priority; H schedules override L schedules and low priority operator requests.

This priority is checked in case two goals get into conflict. The hierarchy of requests and priorities is implemented as follows:

```
Request from         wins over   request from
------------         ---------   ------------

Service period high  ========>   service period low, INGREQ low
INGREQ low           ========>   service period low
INGREQ high          ========>   service period high, service period low,
                                                      INGREQ low
```

Note:
There are other methods of generating requests within SA z/OS. For a complete description of these methods and the implemented hierarchy of their priorities refer to the documentation of the SA z/OS concepts in *IBM Tivoli System Automation for z/OS User's Guide*.

When defining a service period keep in mind that the startup time is the time at which SA z/OS is to begin startup processing. It is not the time at which the resource will be available. The time until availability depends on how long the startup of an application or an application group takes.

According to the goal driven concept, a vote to start a resource is generated from a service period that you enter into an in a UP service window. A vote to stop a resource is generated from each service period that you enter into a DOWN service window. These votes have to compete with other votes that have been probably issued (for example, due to operator request). At the end of the service period, these votes are removed from SA z/OS. An end of an UP service window does not mean implicitly the start of a DOWN service window.

For more information about the hierarchy of service windows or the syntax of FROM and TO times, see the online help for the product.

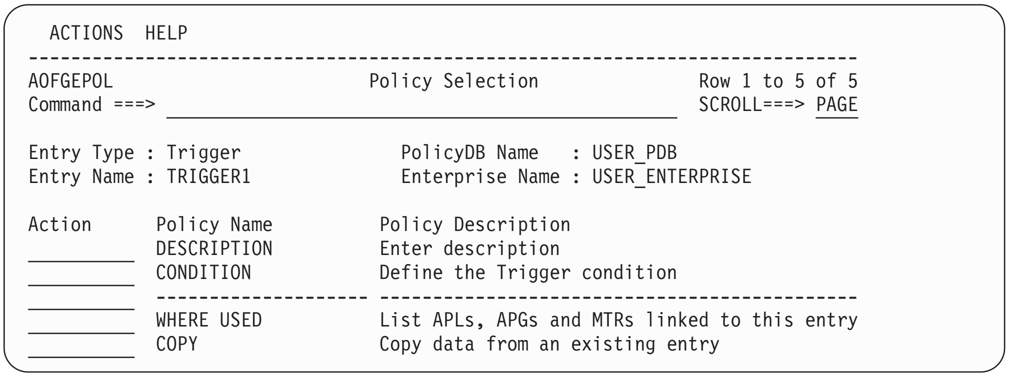
If you later change a service period that is already active, and the service period is activated again, this may lead to conflicts and mismatches with overrides that have been made previously. So if you want to change a service period, check your existing NetView overrides that use this service period beforehand, so that no override is outside the new service period (see also the INGSCHED command, described in *IBM Tivoli System Automation for z/OS Operator's Commands*).

Triggers Policy Object

In combination with events, triggers are used to control the starting and stopping of resources beyond what is done with specified goals. You use events and triggers if external conditions must be fulfilled (such as a tape mount, for example).

Policy Items for Triggers

Figure 165 shows the policy items that are displayed when a trigger is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection              Row 1 to 5 of 5
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Trigger               PolicyDB Name   : USER_PDB
Entry Name : TRIGGER1              Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  CONDITION            Define the Trigger condition
__________  -------------------- --------------------------------------------
__________  WHERE USED           List APLs, APGs and MTRs linked to this entry
__________  COPY                 Copy data from an existing entry
```

Figure 165. Policy Selection Panel for Triggers

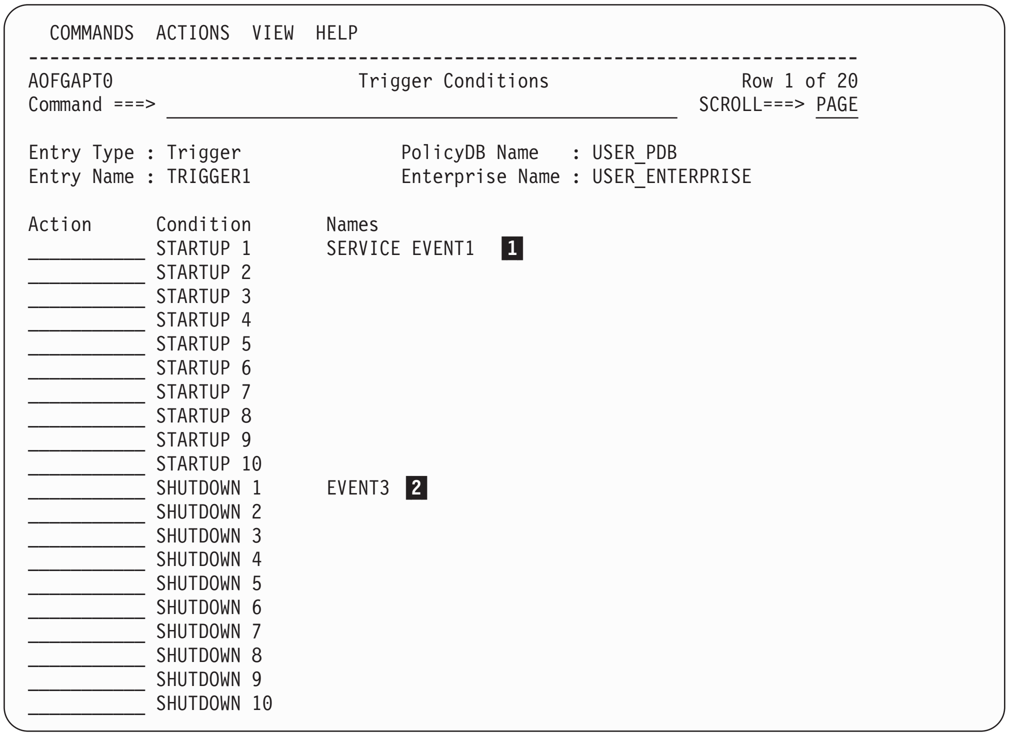
The unique policy item for a trigger is:

CONDITION
This policy item allows you to define the startup and shutdown conditions for this trigger. You need to link the trigger with its defined conditions to Applications, Application Groups, or Monitor Resources.

CONDITION Policy Item

With the CONDITION policy item you can:

1. Define up to ten startup conditions for the application that you want to connect to the trigger (**1**), see "Defining STARTUP and SHUTDOWN Conditions for a Trigger" on page 240.
2. Define up to ten shutdown conditions for the application that you want to connect to the trigger (**2**), see "Defining STARTUP and SHUTDOWN Conditions for a Trigger" on page 240.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGAPT0                      Trigger Conditions                  Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Trigger              PolicyDB Name   : USER_PDB
Entry Name : TRIGGER1             Enterprise Name : USER_ENTERPRISE

Action      Condition       Names
___________ STARTUP 1       SERVICE EVENT1  1
___________ STARTUP 2
___________ STARTUP 3
___________ STARTUP 4
___________ STARTUP 5
___________ STARTUP 6
___________ STARTUP 7
___________ STARTUP 8
___________ STARTUP 9
___________ STARTUP 10
___________ SHUTDOWN 1      EVENT3  2
___________ SHUTDOWN 2
___________ SHUTDOWN 3
___________ SHUTDOWN 4
___________ SHUTDOWN 5
___________ SHUTDOWN 6
___________ SHUTDOWN 7
___________ SHUTDOWN 8
___________ SHUTDOWN 9
___________ SHUTDOWN 10
```

Figure 166. Trigger Conditions Panel

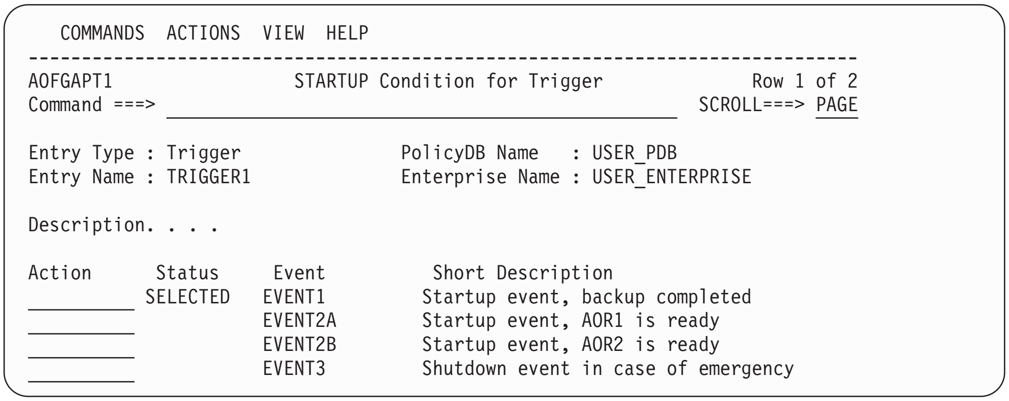
Note:

When you define conditions for a new trigger you get an empty Trigger Conditions panel. The entries in Figure 166 are only visible after you have finished the tasks described in the subsequent sections.

- A maximum number of **20** condition names (service period and events) can be selected for each condition.
- The overall number of different events in a trigger is limited to **28**.
- An event can be specified either in STARTUP conditions or in SHUTDOWN conditions of a particular trigger, however, not in both types of conditions simultaneously.

Defining STARTUP and SHUTDOWN Conditions for a Trigger

If you type an `S` in front of a `STARTUP` condition on the panel shown in Figure 166 (**1**), SA z/OS displays another panel like the one shown in Figure 167 on page 241.



```
   COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGAPT1                 STARTUP Condition for Trigger            Row 1 of 2
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Trigger              PolicyDB Name   : USER_PDB
Entry Name : TRIGGER1             Enterprise Name : USER_ENTERPRISE

Description. . . .

Action      Status     Event         Short Description
__________ SELECTED   EVENT1         Startup event, backup completed
__________            EVENT2A        Startup event, AOR1 is ready
__________            EVENT2B        Startup event, AOR2 is ready
__________            EVENT3         Shutdown event in case of emergency
```

Figure 167. STARTUP Condition for Trigger Panel

Figure 167 displays a list of events where you can select *one or more* events that should trigger the startup of an application. Typing an S will select an event and set its status to SELECTED. All selected events will be added to the currently selected startup condition.

- You can enter a description for the startup condition.
- Entering *YES* on the *Service Period* entry field connects the already selected service period to this condition. In this scenario, you can only connect service period *SERVP1* to any of the conditions for *TRIGGER1*. Therefore, enter *YES* on the *Service Period* entry field only for *STARTUP 1*.

If you type an S in front of a SHUTDOWN condition on the panel shown in Figure 166 on page 240 (**2**), SA z/OS displays another panel similar to the one shown in Figure 167. The panel title in the panel displayed will now read SHUTDOWN Condition for Trigger and will contain a list of events which you can add to or remove from the currently selected shutdown condition.

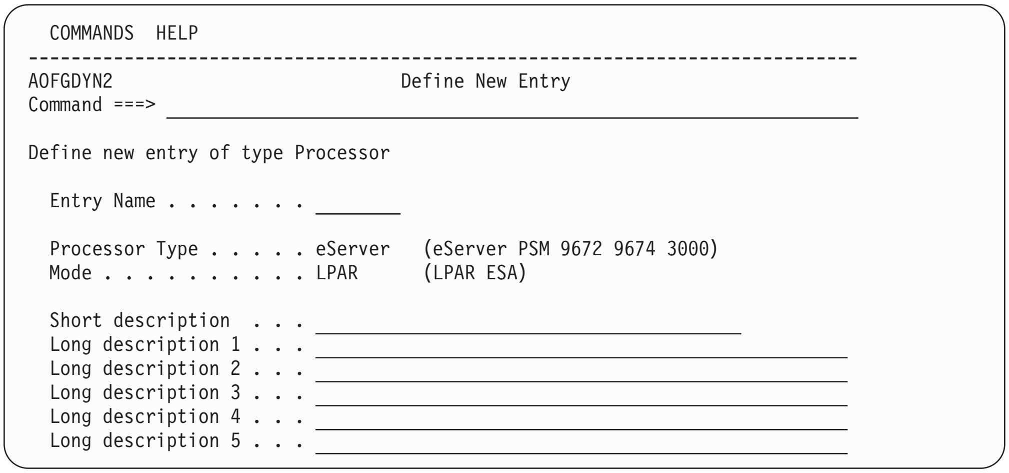
The *Service Period* input field is inhibited when no service period has been selected for the trigger.

Processor Policy Object

Use the Processor Policy Object to define and control your processor hardware in the sysplex.

Creating a New Processor

How to create new policy objects in general is described in “How to Create New Policy Objects” on page 37. Figure 168 on page 242 shows the panel for defining a new Processor policy object.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYN2                           Define New Entry
Command ===> _________________________________________________________________

Define new entry of type Processor

  Entry Name . . . . . . . ________

  Processor Type . . . . . eServer   (eServer PSM 9672 9674 3000)
  Mode . . . . . . . . . . LPAR      (LPAR ESA)

  Short description  . . . ________________________________________
  Long description 1 . . . __________________________________________________
  Long description 2 . . . __________________________________________________
  Long description 3 . . . __________________________________________________
  Long description 4 . . . __________________________________________________
  Long description 5 . . . __________________________________________________
```

Figure 168. Defining a New SA z/OS Processor Entry

Name The processor name must have 1 to 8 alphanumeric characters and must begin with an alphabetic character, #, $, or @. The name must be different from the names of all groups, subgroups, systems (including the processor operations names), and processors defined in the current policy database.

Processor Type
Specifies the processor machine type of the target hardware. Select a name from the list shown to specify the processor type.

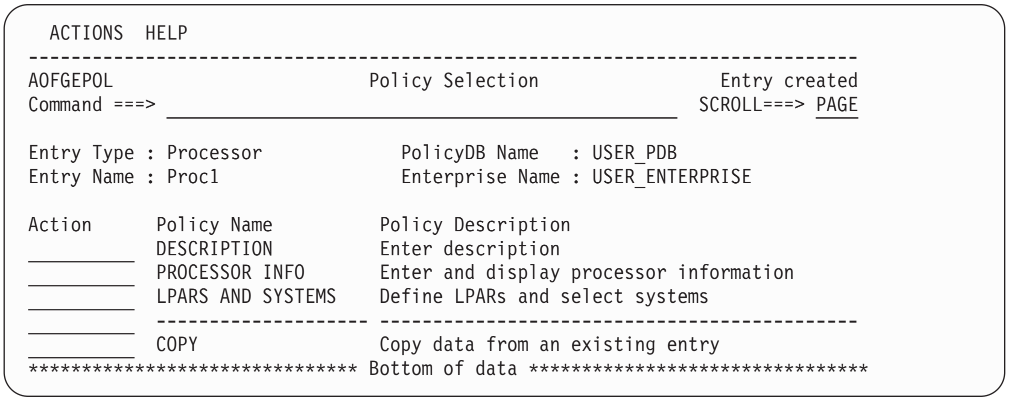
Use the generic type name eServer™ to specify any zSeries or 390-CMOS processor including those shown in the list. The actual processor type is determined at SA z/OS processor operations run time when a target system associated with the target hardware is initialized.

Use the type PSM to specify a ProcOps Service Machine on VM as 'virtual hardware' for VM guest systems. The default is eServer. For more details about the PSM, see *IBM Tivoli System Automation for z/OS Customizing and Programming*.

Mode Specifies the operating mode of the processor hardware as defined in the Reset Profile that is used for hardware activation. For zSeries, 390-CMOS in logical partition mode, and PSM processors specify LPAR.

Policy Items for Processors

The following figure shows the Policy Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection               Entry created
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Processor            PolicyDB Name   : USER_PDB
Entry Name : Proc1                Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  PROCESSOR INFO       Enter and display processor information
__________  LPARS AND SYSTEMS    Define LPARs and select systems
__________  -------------------- ---------------------------------------------
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

Figure 169. Policy Selection Panel for Processor

The unique policy items for the Processor policy object are:

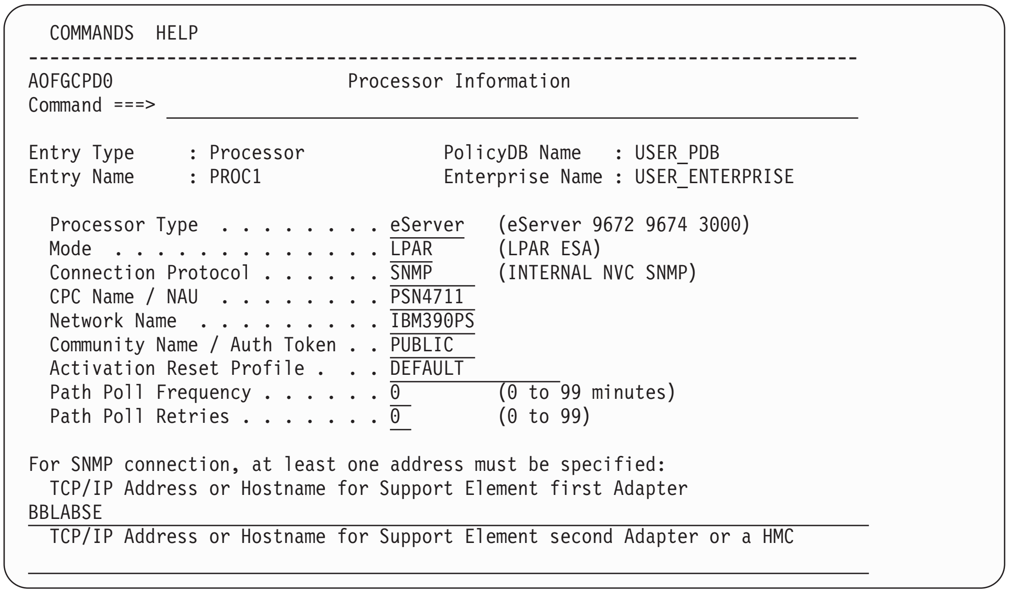
PROCESSOR INFO
Using this policy, you can specify the processor type and mode, the connection protocol and network information.

LPARS AND SYSTEMS
This policy is displayed for processors with mode LPAR.

SYSTEMS
This policy is displayed for processors with mode ESA. Use this policy to select a system.

PROCESSOR INFO Policy Item

A panel similar to Figure 170 is displayed if you select the PROCESSOR INFO policy item from the Policy Selection panel for Processors.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGCPD0                     Processor Information
Command ===>

Entry Type     : Processor             PolicyDB Name   : USER_PDB
Entry Name     : PROC1                 Enterprise Name : USER_ENTERPRISE

  Processor Type  . . . . . . . . . eServer   (eServer 9672 9674 3000)
  Mode  . . . . . . . . . . . . . . LPAR      (LPAR ESA)
  Connection Protocol . . . . . . . SNMP      (INTERNAL NVC SNMP)
  CPC Name / NAU  . . . . . . . . . PSN4711
  Network Name  . . . . . . . . . . IBM390PS
  Community Name / Auth Token . . . PUBLIC
  Activation Reset Profile .  . . . DEFAULT
  Path Poll Frequency . . . . . . . 0         (0 to 99 minutes)
  Path Poll Retries . . . . . . . . 0         (0 to 99)

For SNMP connection, at least one address must be specified:
  TCP/IP Address or Hostname for Support Element first Adapter
BBLABSE
  TCP/IP Address or Hostname for Support Element second Adapter or a HMC
```

Figure 170. Processor Information Panel

On the Processor Information panel you can specify the following definitions so that the automation knows where the system is located to send the commands to the appropriate support element.

Processor Type
Specifies the processor machine type of the target hardware. Select a name from the list to specify the processor type. Use the generic type name eServer to specify any zSeries or 390-CMOS processor including those shown in the list. The actual processor type is determined at SA z/OS processor operations runtime when a target system associated with the target hardware is initialized. This field is required.

Mode Specifies the operating mode of the processor hardware as defined in the Reset Profile used for hardware activation. This field is required.

Connection Protocol
Indicates the protocol used for communication with the processor:

INTERNAL
Connection via the operating system.

NVC Connection via SNA network.

SNMP
Connection via TCP/IP.

A processor with connection INTERNAL is not built into the processor operations control file. It is used for Parallel Sysplex® automation in combination with applications of type IMAGE. For more details, see *IBM Tivoli System Automation for z/OS Customizing and Programming*. With these definitions the automation knows where the system is located to send the commands to the appropriate support element.

CPC Name / NAU
Indicates the network addressable unit of the support element within the network. This field is required for all connections that are supported (INTERNAL, NVC, and SNMP).

Network Name
Indicates the name of the SNA network that the support element is located in. This field is required for all connections that are supported (INTERNAL, NVC, and SNMP). For SNMP connections, the central processor complex name of the support element consists of the Network Name and CPC Name.

Community Name / Auth Token
For SNMP and INTERNAL connections, this indicates the community name of the support element. For NVC connections, this indicates the authorization token. Both community name and authorization token are defined at the support element, and are used to validate command authorization. This field is required.

Activation Reset Profile
Indicates the Reset profile name to use when activating the target hardware. This profile name must match with the Reset profile name defined at the target hardware support element. This field is optional. The Reset profile name specified here will be used by ProcOps common command processing if the CNAME keyword is omitted from the ACTIVATE or LOAD command, and use of a Reset profile is appropriate.

If no Reset profile name is specified here, and the CNAME keyword is not used on the LOAD or ACTIVATE command, ProcOps omits the CNAME parameter from the command sent to the support element. In this case, the support element uses its default Reset profile.

Path Poll Frequency
Enter how often (in minutes) SA z/OS processor operations should poll the support element for status. Valid values are 0 to 99 minutes. Specify 0 to deactivate polling.

Path Poll Retries
Enter the number of attempts to retry that should occur before SA z/OS processor operations notifies the operator that the connection failed. Valid values are 0 to 99 retries.

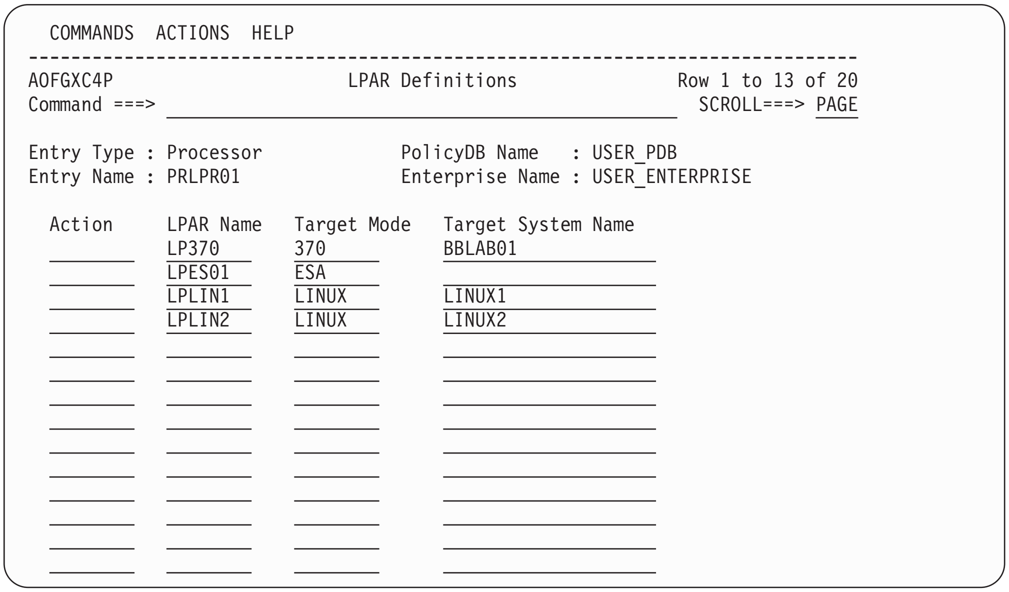
TCP/IP Address or Hostname for Support Element first Adapter/second Adapter or a HMC
TCP/IP Address or Hostname indicates TCP/IP address or symbolic hostname for communication with the target hardware.

At least one address is required for SNMP connection. If both are specified, the support element first adapter address has preference and the second adapter address can be used to recover from adapter failures. Use the ProcOps focal point command ISQIPSWT to perform the switch from the first to the second adapter address. Depending on the Support Element version, a second adapter with an alternate address may not be supported.

An HMC address alone may be considered if multiple processors should be controlled via one single connection.

LPARS AND SYSTEMS Policy Item

If you have created a Processor policy object with LPAR mode, the policy item LPARS AND SYSTEMS is available, and has the LPAR Definitions panel shown in Figure 171 associated with it.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGXC4P                      LPAR Definitions              Row 1 to 13 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Processor            PolicyDB Name   : USER_PDB
Entry Name : PRLPR01              Enterprise Name : USER_ENTERPRISE

  Action     LPAR Name   Target Mode   Target System Name
  ________   LP370       370           BBLAB01
  ________   LPES01      ESA           ____________________
  ________   LPLIN1      LINUX         LINUX1
  ________   LPLIN2      LINUX         LINUX2
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
  ________   ________    ________      ____________________
```

Figure 171. LPAR Definitions Panel

You can specify the following:

Action
Action allows you to remove LPAR definitions and to invoke the system selection panel:

- DELETE (D) deletes the current LPAR definition. Confirmation on a separate panel is not required.
- SYSTEM (S) displays a selection list of all defined systems allowing a system to link to the current LPAR.

LPAR Name
Specifies the name of the logical partition. Duplicate LPAR names without Target System Name are wiped out when the panel is left. However, it is possible to link different systems to one LPAR, so duplicate LPAR names are accepted when they have links to different systems. Linking one system to multiple LPARs is not accepted.

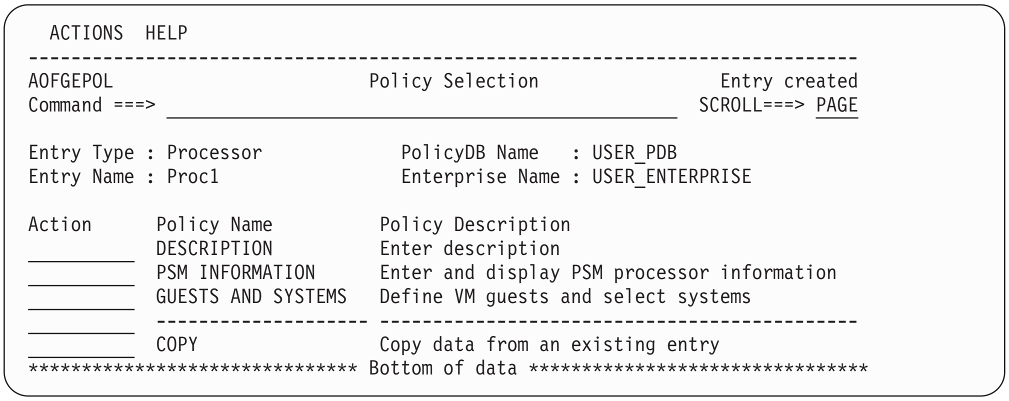
Target Mode
Specifies the mode of the system that is supposed to run in this LPAR. The Target Mode LINUX is used for hardware that supports LPARs for Linux only, where a reduced instruction set is provided and therefore cannot run MVS or VM. For a Linux system specify LINUX if the processor hardware is configured in 'LINUX Only' mode, otherwise specify ESA.

Target System Name
You can specify a system entry name, which establishes a "link" of this

system to the partition. Any name is allowed here, it is not required that the system entry has already been defined. During build, such an undefined system is treated as not specified. If the system is already defined as SYS entry, it is checked that it is a valid selection according to the rules described in the following section for entry type SYS. Otherwise, an error message is displayed.

Policy Items for PSM Processors

Figure 172 shows the Policy Selection panel for PSM processors.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                 Entry created
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Processor             PolicyDB Name   : USER_PDB
Entry Name : Proc1                 Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  PSM INFORMATION      Enter and display PSM processor information
__________  GUESTS AND SYSTEMS   Define VM guests and select systems
__________  -------------------- --------------------------------------------
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

Figure 172. Policy Selection Panel for PSM Processors

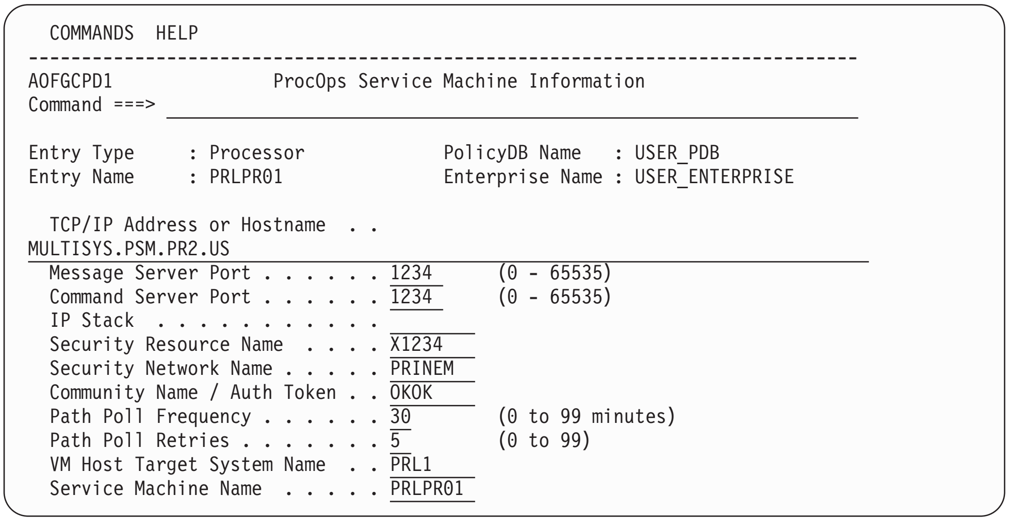
The unique policy items for the Processor policy object are:

PSM INFORMATION
Using this policy, you can specify the processor type and mode, the connection protocol and network information.

GUESTS AND SYSTEMS
This policy allows you to define VM guests and select systems. For more details, see “GUESTS AND SYSTEMS Policy Item” on page 248.

PSM INFORMATION Policy Item

A panel similar to Figure 173 on page 247 is displayed if you select the PSM INFORMATION policy item from the Policy Selection panel for PSM Processors.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGCPD1              ProcOps Service Machine Information
Command ===> 

Entry Type     : Processor            PolicyDB Name   : USER_PDB
Entry Name     : PRLPR01              Enterprise Name : USER_ENTERPRISE

  TCP/IP Address or Hostname  . .
MULTISYS.PSM.PR2.US
  Message Server Port . . . . . . 1234      (0 - 65535)
  Command Server Port . . . . . . 1234      (0 - 65535)
  IP Stack  . . . . . . . . . . .
  Security Resource Name  . . . . X1234
  Security Network Name . . . . . PRINEM
  Community Name / Auth Token . . OKOK
  Path Poll Frequency . . . . . . 30        (0 to 99 minutes)
  Path Poll Retries . . . . . . . 5         (0 to 99)
  VM Host Target System Name  . . PRL1
  Service Machine Name  . . . . . PRLPR01
```

Figure 173. ProcOps Service Machine Information Panel

On this panel you can update the following:

TCP/IP Address or Hostname
Indicates TCP/IP address or symbolic host name for communication with the ProcOps Service Machine. This address is required.

Message Server Port
Indicates the port the ProcOps Service Machine Message Server uses for communication with the ProcOps receiver task.

Command Server Port
Indicates the port the ProcOps Service Machine Command Server uses for communication with the ProcOps send program.

IP Stack
Indicates the IP stack that is to be used. If this is not defined, the default IP stack is used.

Security Resource Name
This field corresponds to the CPC Name / NAU name for NVC and INTERNAL connections and is used for SAF checking. This field is required.

Security Network Name
This field corresponds to the Network Name for NVC and INTERNAL connections and is used for SAF checking. This field is required.

Community Name / Auth Token For TCPIP connections
This indicates the authorization token (ProcOps Service Machines are connected only via TCPIP). This field is required.

Path Poll Frequency
Indicates how frequently the support element should be queried for status. Frequency must be specified in whole minutes in a range from 0 to 99. A frequency of 0 means that polling is deactivated. This field is required.

Path Poll Retries
Indicates the number of times query should be retried before notifying the operator. Valid values are 0 to 99 retries. This field is required.

VM Host Target System Name
Indicates the name of the VM host system if it is also defined in ProcOps This field is optional. The number of PSMs per hosting VM system is restricted to 12.

> **Note:**
> The host command ISQXPSM can only be used to operate the PSM via the VM host if this information is provided.

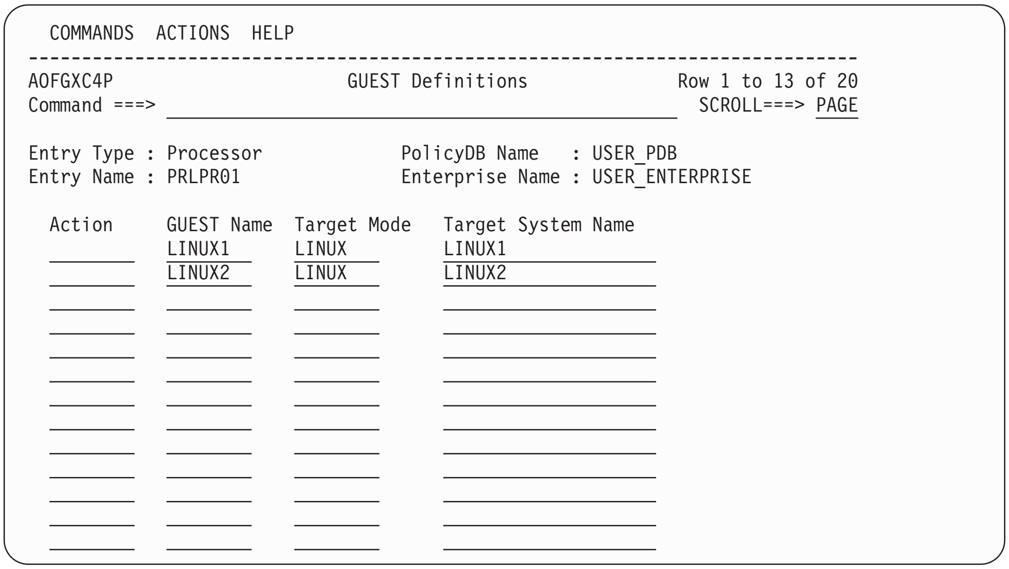
ProcOps Service Machine Name
Indicates the name of the PSM as it is known to the VM host system.

> **Note:**
> The host command ISQXPSM can only be used to operate the PSM via the VM host if this information is provided.

For ProcOps Service Machines, a Mode and Connection Protocol field is not available because the Mode is always LPAR and the Connection Protocol is always TCPIP and neither value can be changed.

GUESTS AND SYSTEMS Policy Item

If you have created a Processor policy object of type PSM, the GUESTS AND SYSTEMS policy item is available, and has the GUEST Definitions panel shown in Figure 174 associated with it.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGXC4P                      GUEST Definitions             Row 1 to 13 of 20
Command ===>                                                   SCROLL===> PAGE

Entry Type : Processor            PolicyDB Name   : USER_PDB
Entry Name : PRLPR01              Enterprise Name : USER_ENTERPRISE

  Action     GUEST Name  Target Mode   Target System Name
             LINUX1      LINUX         LINUX1
             LINUX2      LINUX         LINUX2
```

Figure 174. GUEST Definitions Panel

This panel allows you to define VM GUESTs and to link systems to them. Within ProcOps, VM guest systems are treated like LPARs.

Action
Action allows you to remove VM GUEST definitions and to invoke the system selection panel:

- DELETE (D) deletes the current VM GUEST definition. Confirmation on a separate panel is not required.

- SYSTEM (S) displays a selection list of all defined systems allowing a system to link to the current VM GUEST

GUEST Name
Specifies the name of the VM GUEST. Duplicate VM GUEST names without Target System Name are wiped out when the panel is left. However, it is possible to link different systems to one VM GUEST, so duplicate VM GUEST names are accepted when they have links to different systems. Linking one system to multiple VM GUESTs is not accepted.

Target Mode
Specifies the mode of the system that is supposed to run on this VM GUEST. The Target Mode LINUX is used for hardware that supports VM GUESTs for Linux only, where a reduced instruction set is provided and therefore cannot run MVS or VM.

For a Linux system specify LINUX if the processor hardware is configured in 'LINUX Only' mode, otherwise specify ESA. Valid VM GUEST Target Modes are 370, ESA, and LINUX.

Target System Name
You can specify the name of the system that runs on this VM GUEST. Any name is allowed here; it is not required that the system entry has already been defined. During build, such an undefined system is treated as not specified.

If the system is already defined as a SYS entry, it is checked that it is a valid selection according to the rules described in the following section for entry type SYS. Otherwise, an error message is displayed.

The number of guest systems per PSM is restricted to 60. When a new guest system is defined, the number of guests that have already been defined for the associated PSM is checked. If there are already 60 defined, any further guest definition is rejected.

Monitor Resource Policy Object

The Monitor Resource (MTR) policy object allows you to obtain the health state of an object in two different ways:

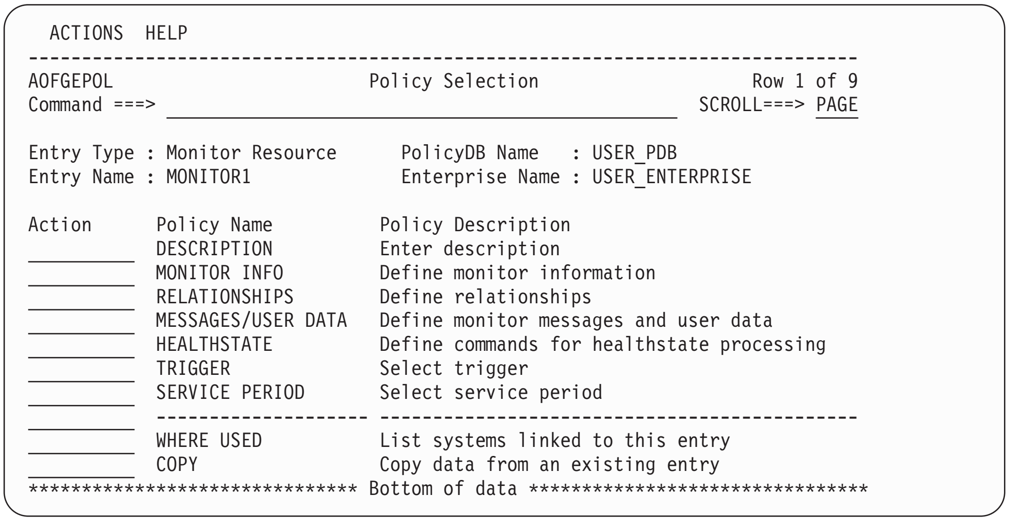
- Actively, by polling, that is, executing a monitoring command periodically
- Passively, by processing events

Monitor resources are connected to application resources (APLs) or application group resources (APGs). The health status of the monitored object is propagated to the APLs or APGs and results in a health status there.

Selecting *Monitor Resources* (MTR) from the Entry Type Selection panel will bring you to the Entry Name Selection panel that lists your current monitor resources, if you have any. From this panel you can either create new monitor resources, or select an entry and action for one of the existing entries.

Policy Items for Monitor Resources

Figure 175 on page 250 shows the policy items that are displayed when an existing Monitor Resources entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                  Row 1 of 9
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Monitor Resource      PolicyDB Name   : USER_PDB
Entry Name : MONITOR1              Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  MONITOR INFO         Define monitor information
__________  RELATIONSHIPS        Define relationships
__________  MESSAGES/USER DATA   Define monitor messages and user data
__________  HEALTHSTATE          Define commands for healthstate processing
__________  TRIGGER              Select trigger
__________  SERVICE PERIOD       Select service period
__________  -------------------- ---------------------------------------------
__________  WHERE USED           List systems linked to this entry
__________  COPY                 Copy data from an existing entry
****************************** Bottom of data ********************************
```

Figure 175. Policy Selection for Monitor Resource

The unique policy items for the Monitor Resource policy object are:

MONITOR INFO
This policy item allows you to make definitions for the monitor resource. Refer to "MONITOR INFO Policy Item" on page 251 for more details.

RELATIONSHIPS
This policy item has the same functionality for entry types Application, ApplicationGroup, and *Monitor Resource*. For each application, application group, or monitor resource you may specify several relationships. They will be used to automate the startup or stopping of those applications, application groups, and monitor resources. If you select this policy item, in the Relationship Selection List panel, you can view existing relationships or define new relationships for the current application, application group, or monitor resource.

You define relationships between dependent resources and supporting resources. In the customization dialog you define relationships always from the point of view of the dependent resource. So the resource that you have currently selected will become the dependent resource, when you define a new relationship.

Refer to "RELATIONSHIPS Policy Item" on page 143 for more information on viewing existing or defining new dependency relationships.

MESSAGES/USER DATA
This policy item allows you to define actions to be taken in response to particular events. Normally, these events are message IDs, but they can also be OMEGAMON exceptions. Refer to "MESSAGES/USER DATA Policy Item" on page 253 for details.

HEALTHSTATE
This policy item allows you to define the processing that takes place when the monitor resource detects a change of the health state of the monitored resource. Refer to "HEALTHSTATE Policy Item" on page 257 and "Event and Trigger Support" on page 25 for more details.

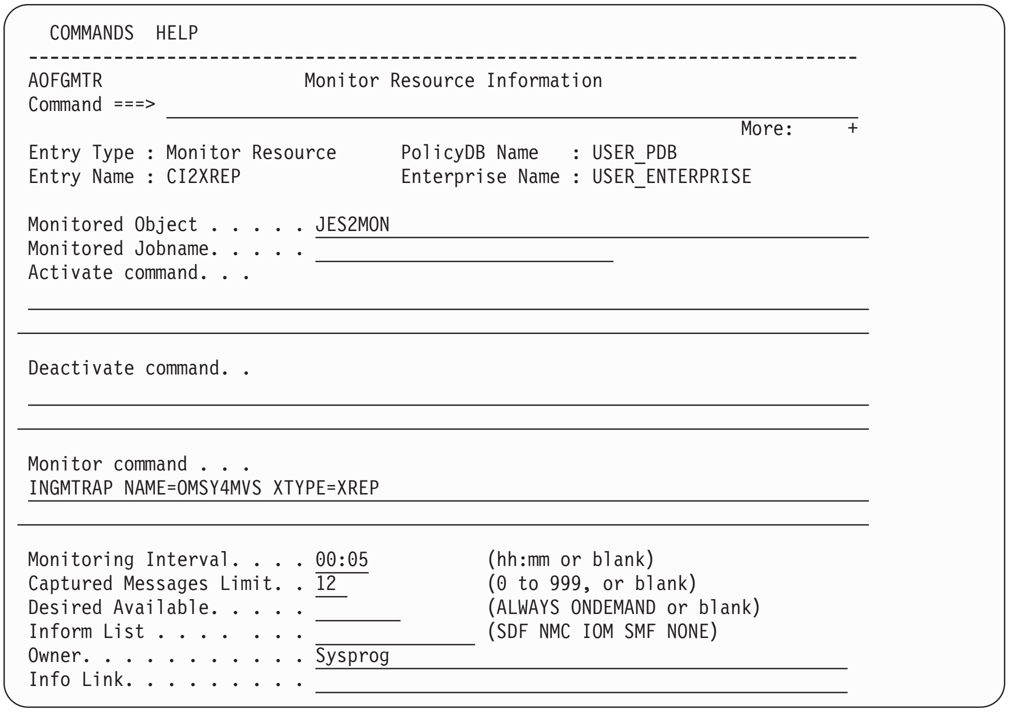
TRIGGER
This policy item allows you to manage the link from the current Monitor Resource entry to the Trigger entry. Each Monitor Resource entry may be

linked to one Trigger entry. Refer to “TRIGGER Policy Item” on page 258 and “Event and Trigger Support” on page 25 for more details.

SERVICE PERIOD
This policy item allows you to link a service period to the monitor resource. Its resources are then automatically started up or shut down in the time frame specified by the service period.

MONITOR INFO Policy Item

If you select the MONITOR INFO policy item from the Policy Selection panel for monitor resources, the Monitor Resource Information panel is displayed, as shown in Figure 176.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMTR                  Monitor Resource Information
Command ===> _________________________________________________________________
                                                                   More:     +
Entry Type : Monitor Resource      PolicyDB Name   : USER_PDB
Entry Name : CI2XREP               Enterprise Name : USER_ENTERPRISE

Monitored Object . . . . . JES2MON_____________________________________________
Monitored Jobname. . . . . ____________________________
Activate command. . .
_______________________________________________________________________________
_______________________________________________________________________________

Deactivate command. .
_______________________________________________________________________________
_______________________________________________________________________________

Monitor command . . .
INGMTRAP NAME=OMSY4MVS XTYPE=XREP______________________________________________
_______________________________________________________________________________

Monitoring Interval. . . . 00:05          (hh:mm or blank)
Captured Messages Limit. . 12_            (0 to 999, or blank)
Desired Available. . . . . ________       (ALWAYS ONDEMAND or blank)
Inform List . . . . . . .  ______________ (SDF NMC IOM SMF NONE)
Owner. . . . . . . . . . . Sysprog___________________________________________
Info Link. . . . . . . . . __________________________________________________
```

Figure 176. Monitor Resource Information Panel

To create a new Monitor Resource entry, provide the following information:

Monitored Object
This is the name of the real object that is being monitored by this monitor resource. It is used to locate the monitor resource that is affected by a real monitor event.

For example, the following values can be used:

OLDS if the monitored object is the IMS Online Log Data Set

RECON if the monitored object is the IMS Recovery Control Data Set

You may include system symbols and system automation symbols. See “Assigning System Automation Symbols (AOCCLONE)” on page 27 for more details.

Note: If you define the same message ID for different monitor resources, the name in the **Monitored Object** field must be identical. This is not required for messages ING150I and ING080I.

Monitored Jobname

This can be used to specify the name of the job that this monitor resource accepts events from. This ensures that only events that are issued from a specific job trigger automation actions for this monitor resource.

For example, if the monitor routine should only react for the IMS Online Log Data Set of a specific IMS, specify the job name here.

You may include system symbols and system automation symbols. See "Assigning System Automation Symbols (AOCCLONE)" on page 27 for more details.

Activate Command

This field is optional and is only needed to establish the monitoring environment.

Deactivate Command

This field is optional and is only needed to establish the monitoring environment.

Monitor Command

This can be any NetView command. If a command and an interval is specified, the command will be executed periodically while the monitor is active.

If a command is specified and no interval is specified, the command will be executed once during the startup phase.

If no command is specified the monitor is a passive monitor. In this case, the health status must be set via the INGMON command (see *IBM Tivoli System Automation for z/OS Programmer's Reference* for more details).

You may include system symbols and system automation symbols but they must be entered in uppercase letters (for example &AOCCLONE1). See "Assigning System Automation Symbols (AOCCLONE)" on page 27 for more details.

Monitoring Interval

Specifies an interval time for periodic execution of the monitor command. An Interval value of 0 will be forced to blank. If an Interval is specified a monitor command must also be specified.

Captured Messages Limit

Specifies the maximum number of history messages that will be saved for the SA z/OS DISPMTR command. If the field is left blank the value can be inherited from the system default definition. If it is also not specified there, a default value of 0 will be used.

Desired Available

Specifies how the desired state of the monitor is determined in the absence of any propagated desiredStatus votes on the monitor.

You can specify the following values:

ALWAYS

The monitor's desired status is set to Available and it is started once its dependencies are satisfied. This is the same behavior that older releases of SA z/OS exhibit.

ONDEMAND

The monitor's desired status is set to Unavailable and it is not be started. For SA z/OS to start the resource it must receive a MakeAvailable vote which is usually propagated along a

HasParent relationship and may be caused by an operator request, a service period, or a Move or Server group.

blank The value can be inherited from the system's default definition.

Note: If nothing is specified nor inherited the default value is ALWAYS

Inform List
This field allows you to specify:

- Where the application is registered to
- Where all status changes are propagated to

Valid receivers are SDF, NMC, IOM, and SMF. You can specify more than one receiver. The list of values can be blank or comma separated.

Neither system automation symbols (AOCCLONEs) nor system symbols are allowed.

If the field is left blank the value is inherited from the systems default definition (see “System Defaults Policy Object” on page 288).

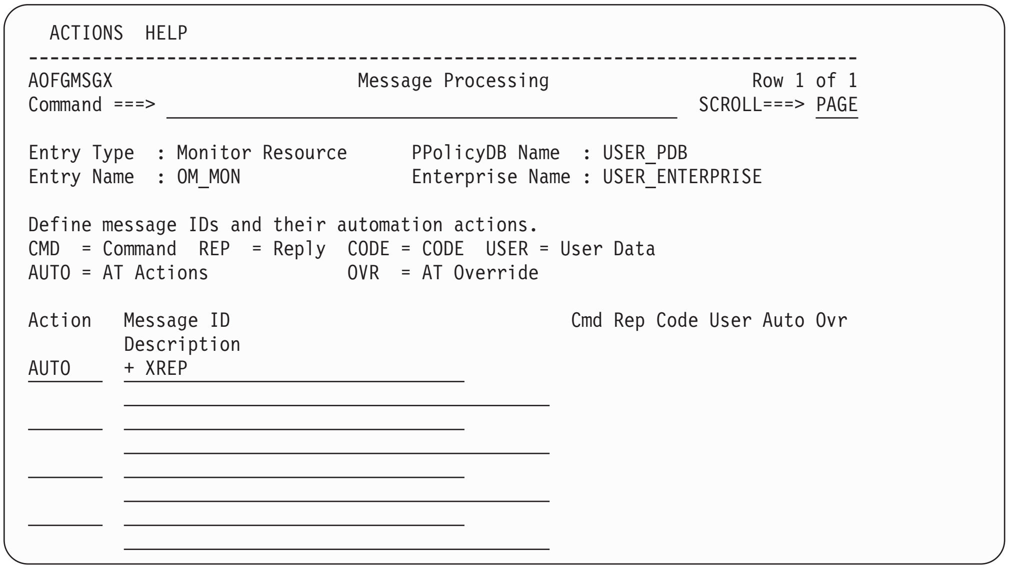
If the resource should not be registered at all, specify NONE. This will prevent inheritance from the systems default definitions.

Owner
Specifies information for the operator about who to contact in case of error.

Info Link
This field can be used to specify a location (for example, a URL) where additional information about the application can be found.

MESSAGES/USER DATA Policy Item

In entry type MTR, the MESSAGES/USER DATA policy item allows you to specify OMEGAMON exceptions and messages in the Message ID field, as shown in Figure 177. Note that the ‘+’ sign and the imbedded blank must be specified to distinguish the exception from regular messages.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGMSGX                      Message Processing                    Row 1 of 1
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type  : Monitor Resource      PPolicyDB Name  : USER_PDB
Entry Name  : OM_MON                Enterprise Name : USER_ENTERPRISE

Define message IDs and their automation actions.
CMD  = Command  REP  = Reply  CODE = CODE  USER = User Data
AUTO = AT Actions             OVR  = AT Override

Action   Message ID                                Cmd Rep Code User Auto Ovr
         Description
AUTO     + XREP
```

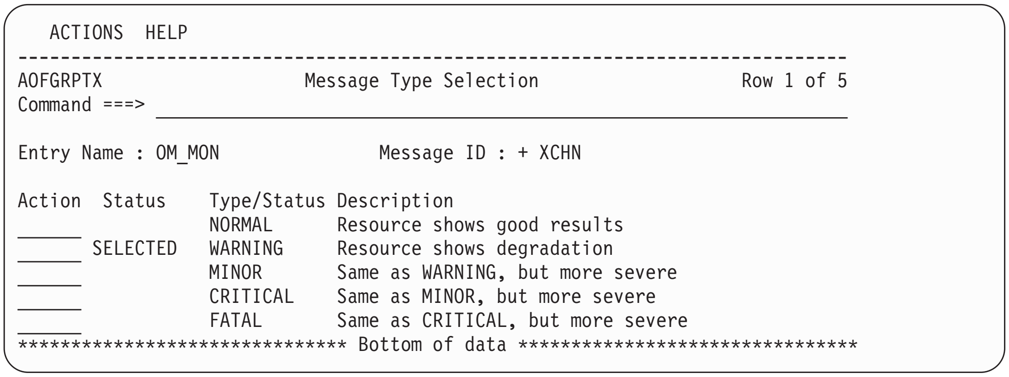
Figure 177. Message Processing Panel with OMEGAMON Exception

With an OMEGAMON exception in the Message ID field, actions CMD, CODE, USER, and AUTO can be used to specify commands, codes, user data, and health states, respectively. This results in a default automation table entry that is processed whenever message ING080I is issued on behalf of the subject monitor and exception. If the defaults provided by the customization dialog do not meet the installation's needs an override action can be selected to handcraft the AT entry.

On the command processing panel (accessed via action CMD), a set of one or more commands can be specified to react to the exception represented by ING080I

You can also define generic key/value pairs (USER DATA) that are available to the called command. SA z/OS provides a predefined key, DISABLETIME, which you can use to determine the time that processing for this particular exception will be disabled. The commands that are specified will be issued, but will only be re-issued if the exception still exists after the time specified with DISABLETIME. This is useful if the recovery for an exception takes a while and no subsequent parallel recovery actions should be taken while the initial recovery is still running.

The AUTO action allows you to select the health state of the monitor whenever this exception tripped and message ING080I was issued. When AUTO is specified, a panel similar to Figure 178 is displayed.



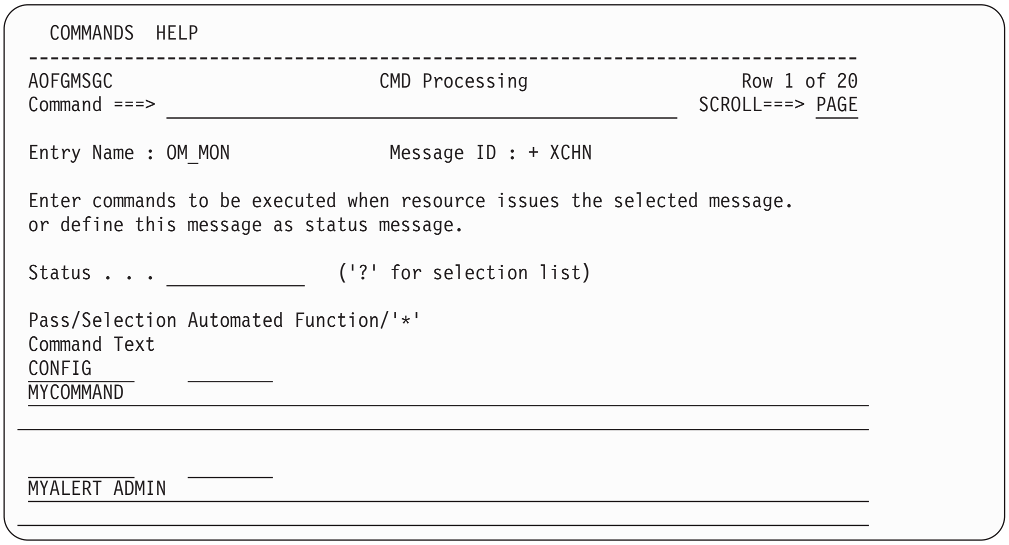
```
   ACTIONS  HELP
------------------------------------------------------------------------------
AOFGRPTX                   Message Type Selection                  Row 1 of 5
Command ===> _________________________________________________________________

Entry Name : OM_MON              Message ID : + XCHN

Action  Status    Type/Status Description
______            NORMAL      Resource shows good results
______ SELECTED   WARNING     Resource shows degradation
______            MINOR       Same as WARNING, but more severe
______            CRITICAL    Same as MINOR, but more severe
______            FATAL       Same as CRITICAL, but more severe
****************************** Bottom of data ********************************
```

Figure 178. Message Type Selection Panel

Note: The variable MTRNM is available for all exceptions that cause an ING080I message to be created.

The following series of panels illustrates the use of CODE and OVR to manually specify an AT entry with different sets of commands based on attributes contained within the exception message.

As an example, the XCHN exception is used. A code entry is specified to return the value CONFIG whenever channel path 26 is found to be not available. The appropriate reaction is to issue the command MYCMD to fix it. In addition, and for all other channels that are not available, the administrator should be notified. Figure 179 on page 255 shows the input for action CMD.



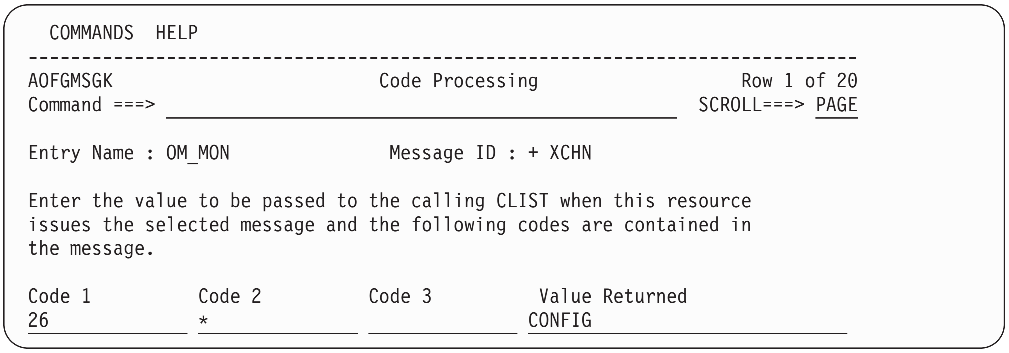
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGC                         CMD Processing                    Row 1 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Name : OM_MON               Message ID : + XCHN

Enter commands to be executed when resource issues the selected message.
or define this message as status message.

Status . . .              ('?' for selection list)

Pass/Selection Automated Function/'*'
Command Text
CONFIG
MYCOMMAND



MYALERT ADMIN

```

Figure 179. CMD Processing Panel

Figure 179 shows two command groups (CONFIG and unnamed) with a total of three commands. To execute the command in the group named CONFIG, a code entry has to be defined that results in value CONFIG. No special action is required to execute the commands in the unnamed group, they will always be issued.

With action CODE on the Message Processing panel of Figure 177 on page 253, a table of codes can be specified that will be evaluated by INGMON to select the commands to issue based on the actual setting of one or more codes. Figure 180 shows an example where the value CONFIG is returned whenever *code1* has a value of 26. In this case, it is channel path 26 because the XCHN exception is being processed.

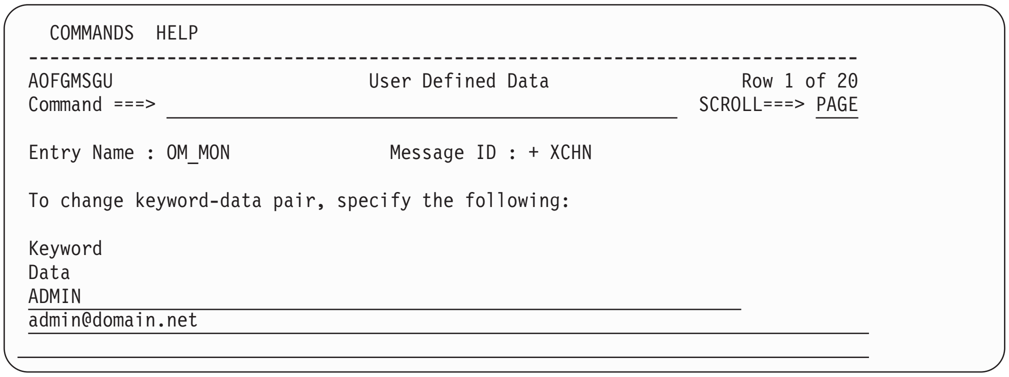
A row is matched whenever the conjunction of *code1*, *code2*, and *code3* evaluates to true. Otherwise, the next row is evaluated for a match. If no match is found, the value returned will be ALWAYS.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGK                         Code Processing                   Row 1 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Name : OM_MON               Message ID : + XCHN

Enter the value to be passed to the calling CLIST when this resource
issues the selected message and the following codes are contained in
the message.

Code 1          Code 2          Code 3           Value Returned
26              *                               CONFIG
```

Figure 180. Code Processing Panel

With action USER on the Message Processing panel of Figure 177 on page 253, a key/value pair can be specified that denotes the administrator to be notified, as shown in Figure 181 on page 256. (There must be a REXX exec that performs this function.)



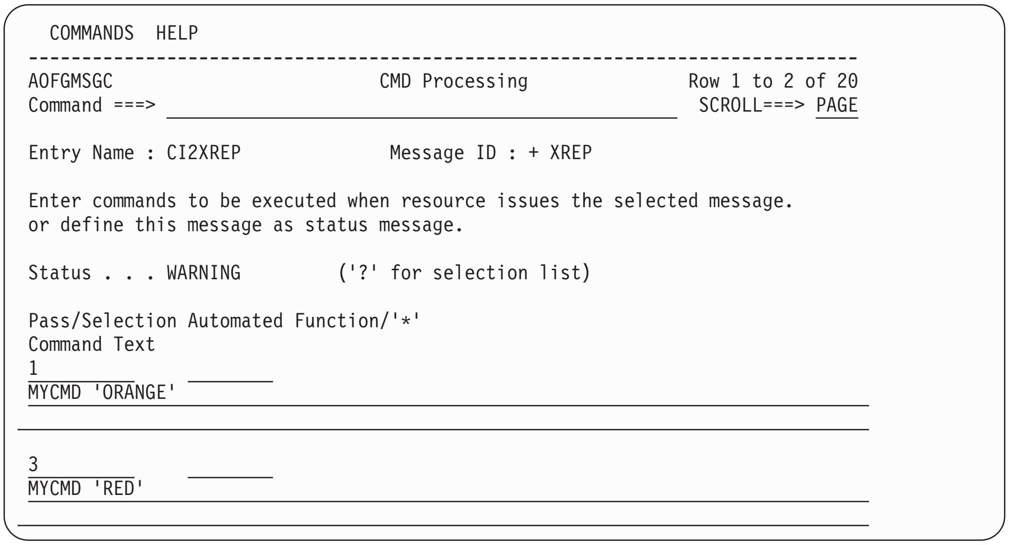
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGU                        User Defined Data                 Row 1 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Name : OM_MON               Message ID : + XCHN

To change keyword-data pair, specify the following:

Keyword
Data
ADMIN
admin@domain.net
```

Figure 181. User Defined Data Panel

You can also use the CMD Processing panel to set up Pass Processing to make multiple attempts to recover from or resolve the exception. PASS counts from 1 to 99 and is incremented for each instance of the message or exception, unless recovery is disabled. In the example in Figure 182, when the number of outstanding replies exceeds the OMEGAMON threshold:

- The first time, issue MYCMD and pass parameter 'ORANGE'
- The second time, do nothing
- The third time, issue MYCMD and pass parameter 'RED'

Note that a blank pass means that the command is always executed.

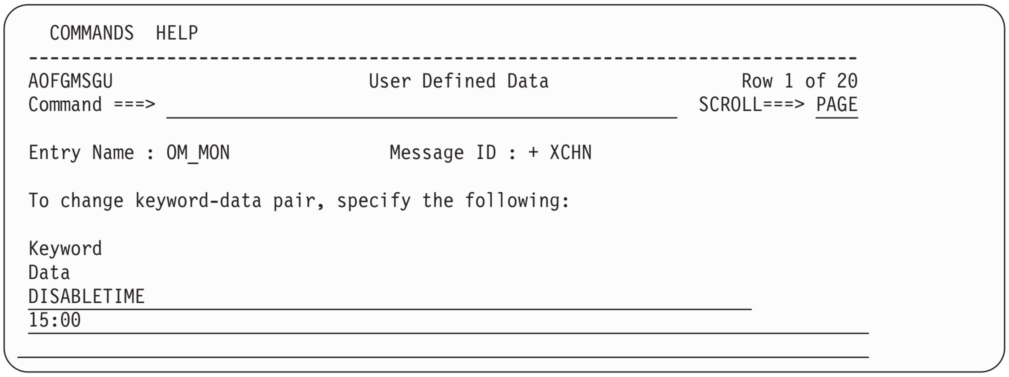


```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGC                         CMD Processing              Row 1 to 2 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Name : CI2XREP              Message ID : + XREP

Enter commands to be executed when resource issues the selected message.
or define this message as status message.

Status . . . WARNING        ('?' for selection list)

Pass/Selection Automated Function/'*'
Command Text
1
MYCMD 'ORANGE'


3
MYCMD 'RED'
```

Figure 182. Monitor Resource CMD Processing Panel - Pass Processing

The USER action can also be used to disable further processing within INGMON using the DISABLETIME keyword, as in Figure 183 on page 257. The data value is the time, which can be specified in the normal NetView format: hh:mm:ss, mm:ss, :ss, or mm..

A DISABLETIME of 0 means that the message processing for this particular combination of monitor and exception is disabled forever. If message processing should be enabled again or prematurely, an INGMON CLEARING request must be issued. It is best issued from within the command that handles the resolution.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGU                       User Defined Data                Row 1 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Name : OM_MON               Message ID : + XCHN

To change keyword-data pair, specify the following:

Keyword
Data
DISABLETIME
15:00
```

Figure 183. User Defined Data Panel

Action OVR on the Message Processing panel of Figure 177 on page 253 needs to be used to manually refine the NetView AT-entry in order to exploit the code entry. SA z/OS generates a default condition and action statement that can be modified by the installation. In the example, the condition statement would be modified to set a variable named MISSCHAN:

```
IF TOKEN(2)='MONITORS/MTR/SYS1' &
   TOKEN(5)='XCHN' &
   TOKEN(10)=MISSCHAN
```

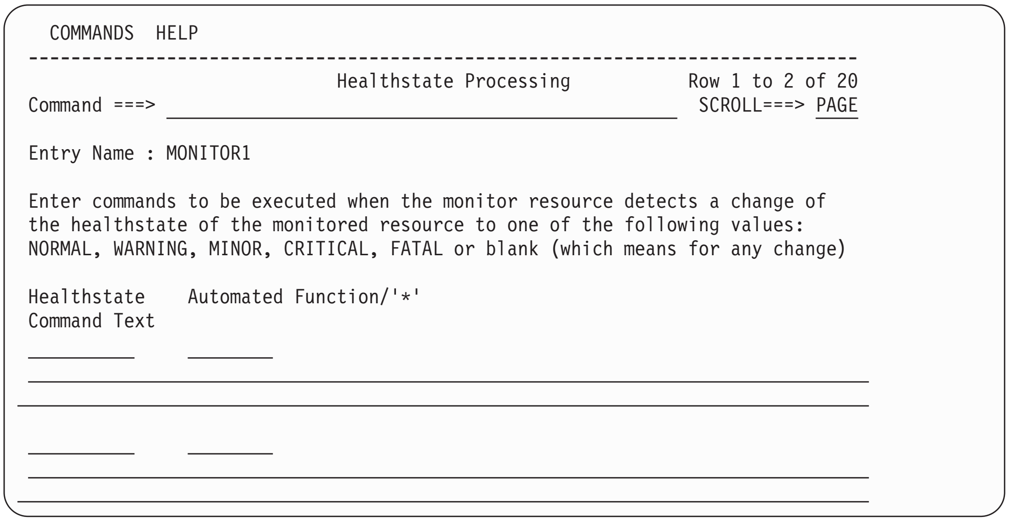
The action statement would be modified to say:

```
EXEC(CMD 'INGMON MONITORS/MTR/SYS1 CODE1='MISSCHAN)
```

OVR can also be used to add the NetView THRESHOLD condition item to the condition statement suggested by SA z/OS. The THRESHOLD condition item counts the number of occurrences within a given time interval and returns a value of true (1) only if the count is equal to or higher than the specified threshold.

HEALTHSTATE Policy Item

If you select the HEALTHSTATE policy item from the Policy Selection panel for monitor resources, the *Healthstate Processing* panel is displayed, as shown in Figure 184.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
                              Healthstate Processing          Row 1 to 2 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Name : MONITOR1

Enter commands to be executed when the monitor resource detects a change of
the healthstate of the monitored resource to one of the following values:
NORMAL, WARNING, MINOR, CRITICAL, FATAL or blank (which means for any change)

Healthstate    Automated Function/'*'
Command Text
__________     ________


__________     ________
```

Figure 184. Healthstate Processing Panel

You can use this panel to enter commands to be executed when the monitor resource detects a change in the healthstate of the monitored resource. The following fields are available:

Healthstate

Defines the health state that the monitored resource must change to in order to initiate the specified commands. Valid values are :

- NORMAL
- WARNING
- MINOR
- CRITICAL
- FATAL
- blank

Leaving this field blank means that the specified commands will be executed in any health state change.

Automated Function

Enter the name of the automated function that the command is to run under. If the field is left blank, the function defaults to the one currently running the command list.

> **Note:**
> Entering a '*' enables return code checking for the specified command. This means that processing continues if the command returns a return code of 0; in all other cases the processing is stopped immediately.

Command Text

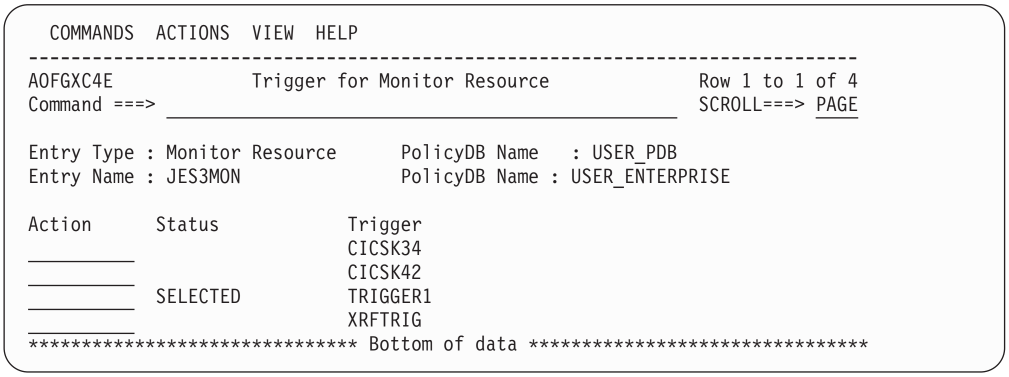
Enter the command or command list to be issued. Any system command must be prefaced with MVS.

A healthstate can be specified on the call to ACFCMD, which will issue the corresponding command or Command Group defined in the CMD entry.

A healthstate of blank can be specified to ensure that the corresponding command is issued regardless of the SEL= parameter included on the ACFCMD call. Thus, for any given healthstate processing, the blank entries will be issued in addition to the entries coded for that particular healthstate.

TRIGGER Policy Item

The Trigger for Monitor Resource panel Figure 185 on page 259 is displayed if you select policy item TRIGGER from the Policy Selection panel for monitor resources. For more details about triggers, refer to “Event and Trigger Support” on page 25.



```
  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGXC4E             Trigger for Monitor Resource              Row 1 to 1 of 4
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Monitor Resource      PolicyDB Name   : USER_PDB
Entry Name : JES3MON               PolicyDB Name : USER_ENTERPRISE

Action      Status            Trigger
__________                    CICSK34
__________                    CICSK42
__________  SELECTED          TRIGGER1
__________                    XRFTRIG
****************************** Bottom of data ********************************
```

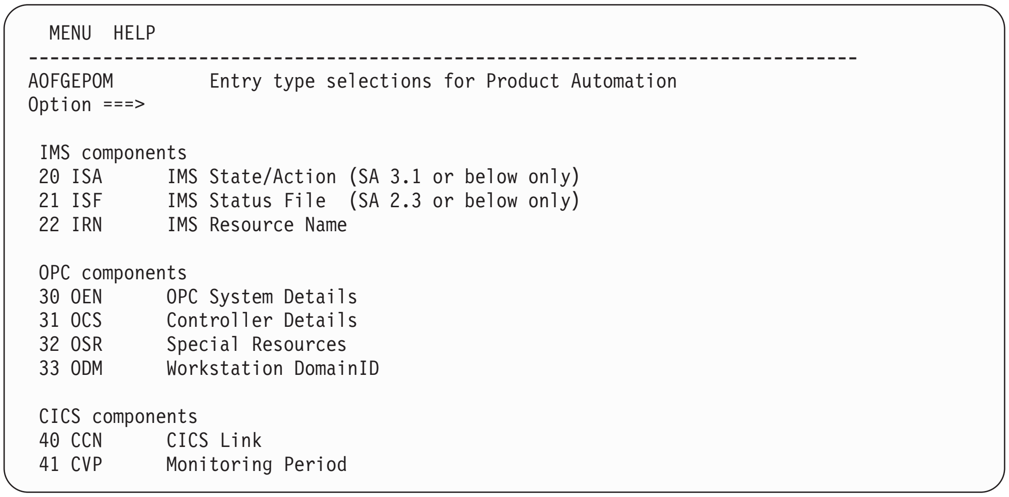
Figure 185. Trigger for Monitor Resource Panel

This panel allows you to manage the link (only one is allowed) between the current Monitor Resource entry and a Trigger entry.

Product Automation Policy Object

Selecting Product Automation (PRD) from the Entry Type Selection panel will bring you to the Entry type selections for Product Automation panel shown in Figure 186 where you can select the product that you want to define automation policy for. You can select from the following products:

- CICS, see "Defining Automation for CICS Components" on page 260
- IMS, see "Defining Automation for IMS Components"
- TWS, see "Defining Automation for TWS Components" on page 260



```
  MENU  HELP
------------------------------------------------------------------------------
AOFGEPOM         Entry type selections for Product Automation
Option ===>

 IMS components
 20 ISA      IMS State/Action (SA 3.1 or below only)
 21 ISF      IMS Status File  (SA 2.3 or below only)
 22 IRN      IMS Resource Name

 OPC components
 30 OEN      OPC System Details
 31 OCS      Controller Details
 32 OSR      Special Resources
 33 ODM      Workstation DomainID

 CICS components
 40 CCN      CICS Link
 41 CVP      Monitoring Period
```

Figure 186. Entry Type Selections for Product Automation Panel

Defining Automation for IMS Components

For IMS components, the following entry types are available:

(ISA) State/Action
With this option, you can define or modify a state/action table for IMS. This is described in more detail in "IMS State/Action Tables Policy Object" on page 261.

Note: This policy item is only for systems that are running earlier releases of SA z/OS.

(ISF) IMS Status file
An IMS Status file policy object is used to define the partner domain and status file in an XRF environment. These objects are linked to System objects. This is described in more detail in "IMS Status Files Policy Object" on page 262.

Note: This policy item is only for systems that are running releases from SA z/OS 2.3 or earlier.

(IRN) IMS resource name
With this option, you can match the XRF RSENAME to the IMS subsystems with which it is associated. This is described in more detail in "IMS Resource Name Policy Object" on page 263.

Defining Automation for TWS Components

For TWS components, the following entry types are available:

(OEN) OPC System details
With this option, you can define certain system wide defaults. Selecting this entry type will bring you to the Entry Name Selection panel for OPC system details where you can create or edit system defaults. You can link an OPC system detail to one or more System policy objects, however, you can link only one OPC system detail to a System. This is described in more detail in "OPC System Detail Policy Object" on page 264.

(OCS) Controller details
With this option, you can link the OPC controller subsystem name with the NetView domain where it is located. Selecting this entry type will bring you to the Entry Name Selection panel for OPC controller details where you can create or edit controller details. You can link an OPC controller detail to one or more System policy objects, however, you can link only one OPC controller detail to a System. This is described in more detail in "OPC Controller Details Policy Object" on page 266.

(OSR) Special resources
Use this option to define special resources that are to be set for an OPC controller. This is described in more detail in "OPC Special Resources Policy Object" on page 268.

(ODM) Workstation domains
Use this option to relate an OPC automatic workstation to a NetView domain ID. This is described in more detail in "OPC Workstation Domains Policy Object" on page 269.

Defining Automation for CICS Components

It is recommended that you use *event-based monitoring* for your CICS link and health monitoring rather than the following policy items. Event-based CICS link and health monitoring is implemented using CICSPlex System Manager (CICSPlex SM or CPSM) objects.

Whenever an event is received from CICSPlex SM, message ING150I is issued. This acts as a trigger that allows you to set the health status of a Monitor Resource based on the monitored object name and job. You can define Monitor Resources for CPSM objects (for example, connections), and codes for message ID ING150I in the MESSAGE/USER DATA policy of the Monitor Resource to map the CPSM severities to valid health states. For more details see the section "Event-Based Monitoring Using Monitor Resources" in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

For CICS components, the following entry types are available:

(CCN) Link definitions
Use this option to define the interregion and intersystem communication (IRC and ISC) links so that they can be monitored by SA z/OS. This is described in more detail in "CICS Link Definitions Policy Object" on page 270.

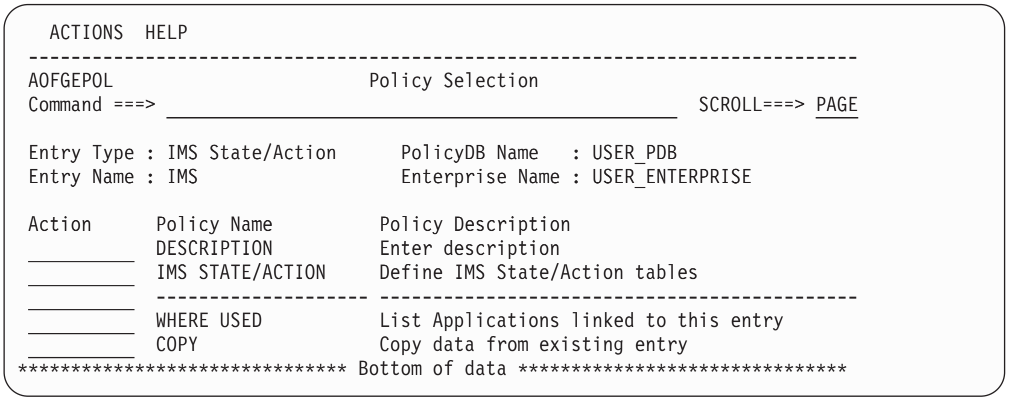
(CVP) Link Monitor periods
With this option, you can define service periods for the link monitoring definitions. Service periods can be related to a CICS link as defined with the **(CCN) Link definition** option. This is described in more detail in "CICS Link Monitor Periods Policy Object" on page 273.

IMS State/Action Tables Policy Object

Figure 187 shows the policy items that are displayed when you want to create a new state/action table for IMS or if you select an existing one from the related Entry Name Selection panel.

Notes:

1. This policy object is only for systems that are running earlier releases of SA z/OS. The definitions are completely ignored in SA z/OS 3.2.
2. For the State/Action function to work for the tables that are shipped with the product, the Entry Name must be IMS.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                         Policy Selection
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : IMS State/Action      PolicyDB Name   : USER_PDB
Entry Name : IMS                   Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  IMS STATE/ACTION     Define IMS State/Action tables
__________  -------------------- ---------------------------------------------
__________  WHERE USED           List Applications linked to this entry
__________  COPY                 Copy data from existing entry
******************************* Bottom of data ********************************
```

Figure 187. Policy Selection Panel for IMS State/Action Table

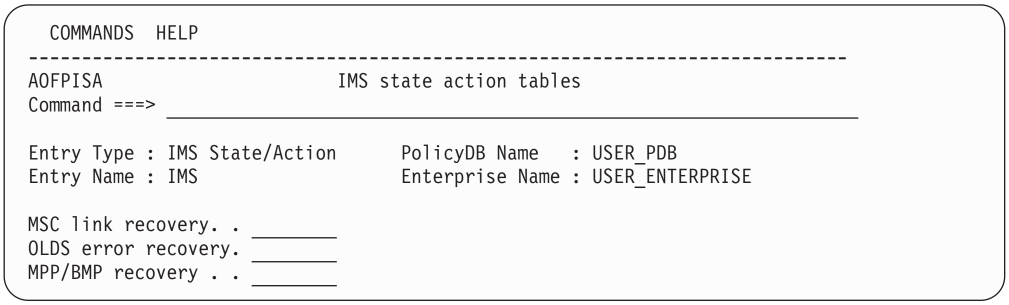
The unique policy item for IMS state/action table is:

IMS STATE/ACTION
This policy item lets you define IMS state/action tables. How to do this is described in more detail in "IMS STATE/ACTION Policy Item."

IMS STATE/ACTION Policy Item

The IMS state action tables panel is displayed if you select the IMS STATE/ACTION policy item from the Policy Selection panel for the IMS State/Action entry type shown in Figure 187. Use this panel to define the state/action tables that are available to your applications. For detailed information on state/action tables please refer to *IBM Tivoli System Automation for z/OS IMS*

Automation Programmer's Reference and Operator's Guide.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISA                      IMS state action tables
Command ===> _________________________________________________________________

Entry Type : IMS State/Action      PolicyDB Name   : USER_PDB
Entry Name : IMS                   Enterprise Name : USER_ENTERPRISE

MSC link recovery. . ________
OLDS error recovery. ________
MPP/BMP recovery . . ________
```

Figure 188. IMS State Action Tables Panel

To define a state/action table, fill in the following fields:

MSC link recovery
The state/action table used for automatic recovery from MSC link errors. The SA z/OS supplied state/action table is EVISS002.

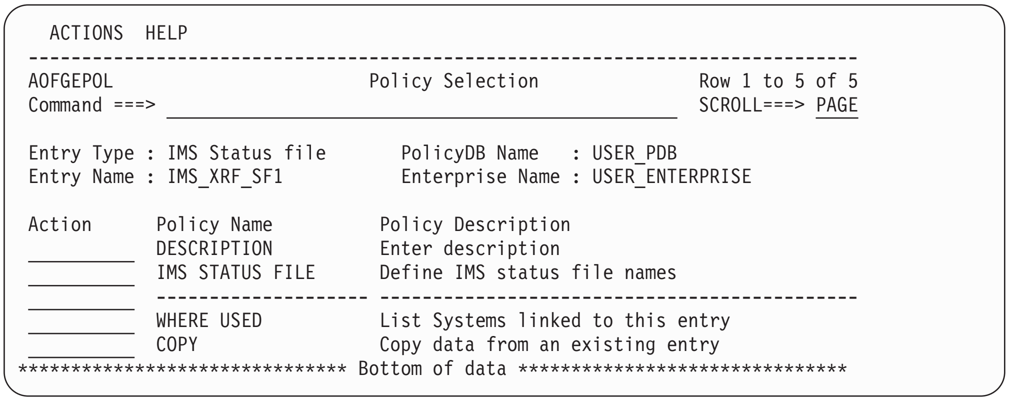
OLDS error recovery
The state/action table used to define procedures for automatic recovery from OLDS errors. The SA z/OS supplied state/action table is EVISS003.

MPP/BMP recovery
The state/action table used to define automatic recovery procedures initiated when an MPP or BMP transaction/program has abended. The SA z/OS supplied state/action table is EVISS005.

IMS Status Files Policy Object

Figure 189 shows the policy items that are displayed when you select an IMS status file from the related Entry Name Selection panel or when you want to create a new IMS status file.

Note: This policy item is only for systems that are running releases from SA z/OS 2.3 or earlier.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection              Row 1 to 5 of 5
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : IMS Status file       PolicyDB Name   : USER_PDB
Entry Name : IMS_XRF_SF1           Enterprise Name : USER_ENTERPRISE

Action       Policy Name          Policy Description
__________   DESCRIPTION          Enter description
__________   IMS STATUS FILE      Define IMS status file names
__________   -------------------- ----------------------------------------
__________   WHERE USED           List Systems linked to this entry
__________   COPY                 Copy data from an existing entry
******************************* Bottom of data *******************************
```

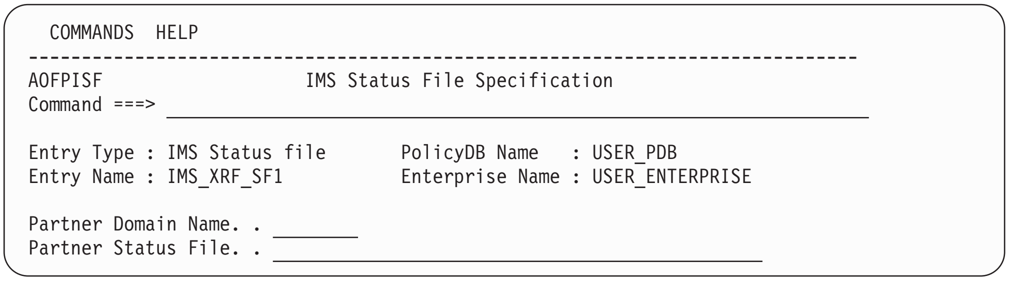
Figure 189. Policy Selection Panel for IMS Status File

The unique policy item for IMS status files is:

IMS STATUS FILE
This policy item lets you define IMS status file names. How to do this is described in more detail in "IMS STATUS FILE Policy Item" on page 263.

IMS STATUS FILE Policy Item

The IMS Status file specification panel is displayed if you select policy item IMS STATUS FILE from the Policy Selection panel for IMS status file shown in Figure 189 on page 262.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPISF                  IMS Status File Specification
Command ===> __________________________________________________________________

Entry Type : IMS Status file       PolicyDB Name   : USER_PDB
Entry Name : IMS_XRF_SF1           Enterprise Name : USER_ENTERPRISE

Partner Domain Name. . ________
Partner Status File. . ______________________________________________
```

Figure 190. IMS Status File Specification Panel

The status file specification describes the partner IMS status file that will be used in an XRF IMS environment. To support an XRF IMS environment, IMS Automation must access the status file of the partner IMS subsystems. This is only required in an XRF environment.

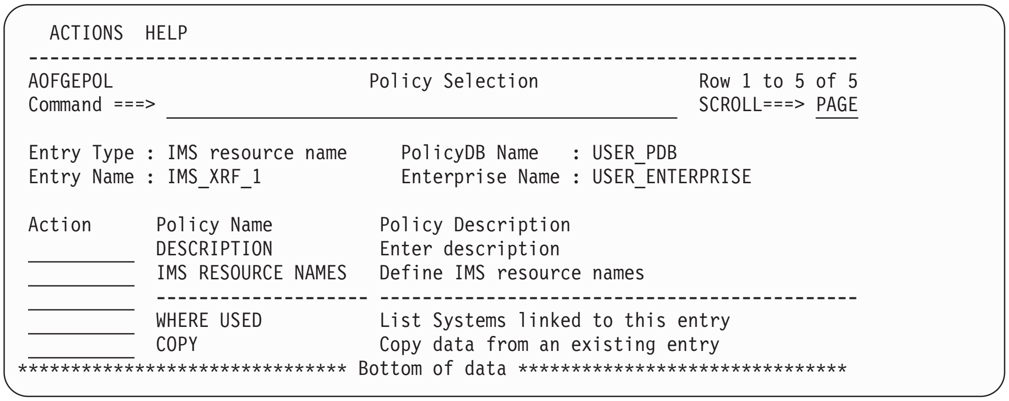
Fill in the following fields:

Partner domain name
This parameter is the NetView domain ID that is controlling the partner IMS subsystem in an XRF environment.

Partner status file
This parameter is the data set name of the status file used by the partner IMS in the partner's domain. This data set is dynamically allocated during SA z/OS initialization.

IMS Resource Name Policy Object

Figure 191 shows the policy items that are displayed when you select an IMS resource name from the related Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection              Row 1 to 5 of 5
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : IMS resource name     PolicyDB Name   : USER_PDB
Entry Name : IMS_XRF_1             Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  IMS RESOURCE NAMES   Define IMS resource names
__________  -------------------- --------------------------------------------
__________  WHERE USED           List Systems linked to this entry
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

Figure 191. Policy Selection Panel for IMS Resource Name

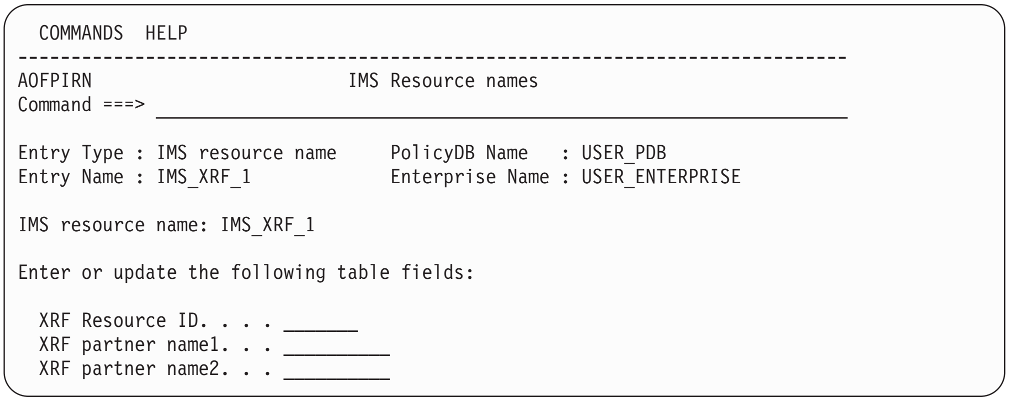
The unique policy item for IMS resource name is:

IMS RESOURCE NAMES
This policy item is used to match the XRF RSENAME to the IMS

subsystems with which it is associated. These entries are linked to SYS entries. For more information, refer to “IMS RESOURCE NAMES Policy Item.”

IMS RESOURCE NAMES Policy Item

The IMS Resource names panel is displayed if you select the IMS RESOURCE NAMES policy item from the Policy Selection panel for IMS Resource names shown in Figure 191 on page 263. Use this panel to define IMS resource names for each XRF IMS complex defined, to match the XRF RSENAME to the IMS subsystems that it is associated with.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIRN                        IMS Resource names
Command ===> _________________________________________________________________

Entry Type : IMS resource name     PolicyDB Name   : USER_PDB
Entry Name : IMS_XRF_1             Enterprise Name : USER_ENTERPRISE

IMS resource name: IMS_XRF_1

Enter or update the following table fields:

  XRF Resource ID. . . . _______
  XRF partner name1. . . __________
  XRF partner name2. . . __________
```

Figure 192. IMS Resource Names Panel

To define IMS resource names, fill in the following fields:

XRF Resource ID
This is the value specified as the RSENAME for the XRF complex as defined in the DFSHSB*xx* member in the IMS PROCLIB.

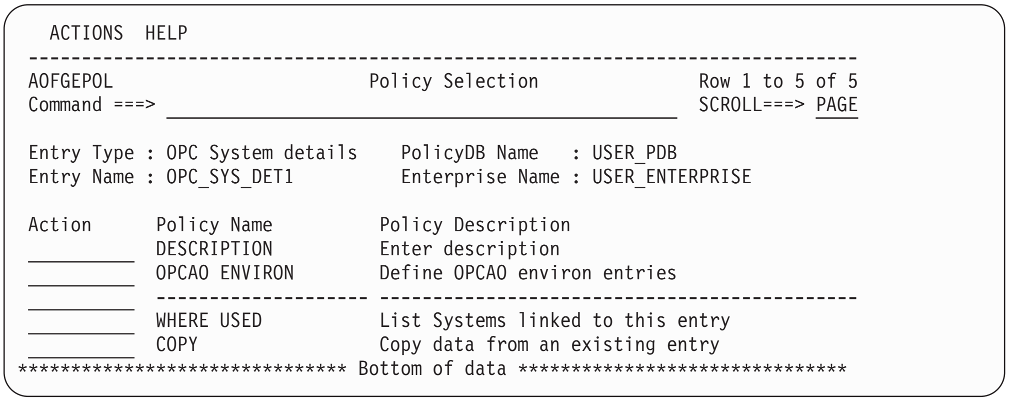
XRF partner name1
This parameter is the SA z/OS subsystem name of one of the IMS control region subsystems that participate in an XRF complex.

XRF partner name2
This parameter is the partner IMS control region subsystem name that participates in this XRF complex.

The order that the IMS subsystems are specified in does not matter.

OPC System Detail Policy Object

Figure 193 on page 265 shows the policy items that are displayed when you select an OPC system detail from the related Entry Name Selection panel or if you want to create a new OPC system detail.



```
  ACTIONS  HELP
 ------------------------------------------------------------------------------
 AOFGEPOL                        Policy Selection             Row 1 to 5 of 5
 Command ===> ________________________________________________  SCROLL===> PAGE

 Entry Type : OPC System details    PolicyDB Name   : USER_PDB
 Entry Name : OPC_SYS_DET1          Enterprise Name : USER_ENTERPRISE

 Action      Policy Name          Policy Description
 __________  DESCRIPTION          Enter description
 __________  OPCAO ENVIRON        Define OPCAO environ entries
 __________  -------------------- ----------------------------------------------
 __________  WHERE USED           List Systems linked to this entry
 __________  COPY                 Copy data from an existing entry
******************************* Bottom of data *******************************
```

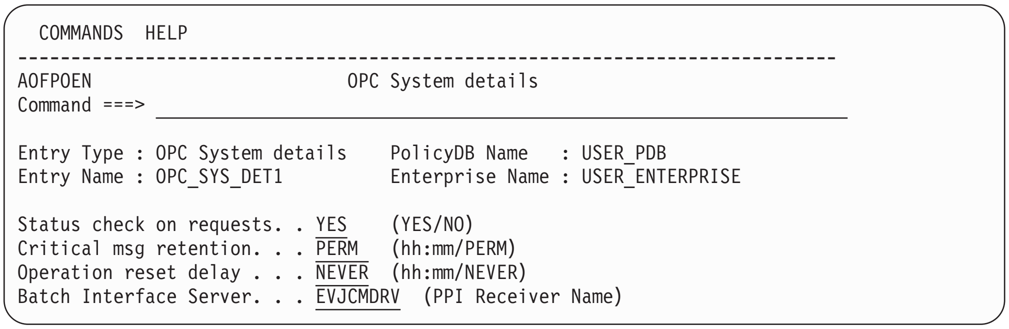
Figure 193. Policy Selection Panel for OPC System Detail

The unique policy item for OPC system details is:

OPCAO ENVIRON
: This policy item checks the status on START and STOP requests, and specifies the retention of critical messages. If you select this policy item, SA z/OS displays the OPC System details panel which is described in more detail in “OPCAO ENVIRON Policy Item.”

OPCAO ENVIRON Policy Item

The OPCAO ENVIRON policy item checks status on START and STOP requests, and specifies retention of critical messages. If you select this policy item from the Policy Selection panel for OPC System Details, SA z/OS displays the panel shown in Figure 194.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPOEN                         OPC System details
Command ===> _________________________________________________________________

Entry Type : OPC System details    PolicyDB Name   : USER_PDB
Entry Name : OPC_SYS_DET1          Enterprise Name : USER_ENTERPRISE

Status check on requests. . YES    (YES/NO)
Critical msg retention. . . PERM   (hh:mm/PERM)
Operation reset delay . . . NEVER  (hh:mm/NEVER)
Batch Interface Server. . . EVJCMDRV  (PPI Receiver Name)
```

Figure 194. OPC System Details Panel

The following fields can be filled in:

Status check
: This entry controls checking of subsystem status on TWS Automation START, STOP, and RECYCLE requests. If not coded, TWS Automation will enforce the rule that START operations will be ended in error if the subsystem is already UP, and likewise for STOP operations if the subsystem is down. Code REQUEST=NO if you wish to have TWS Automation proceed as normal in this case. The default value is YES .

Critical msg retention
: This entry specifies retention of TWS Automation critical messages and TWS error messages. The default is PERM, which means that messages will not be deleted on a timed basis.

Operation reset delay
: This specifies how long the NetView interface to TWS may be unavailable

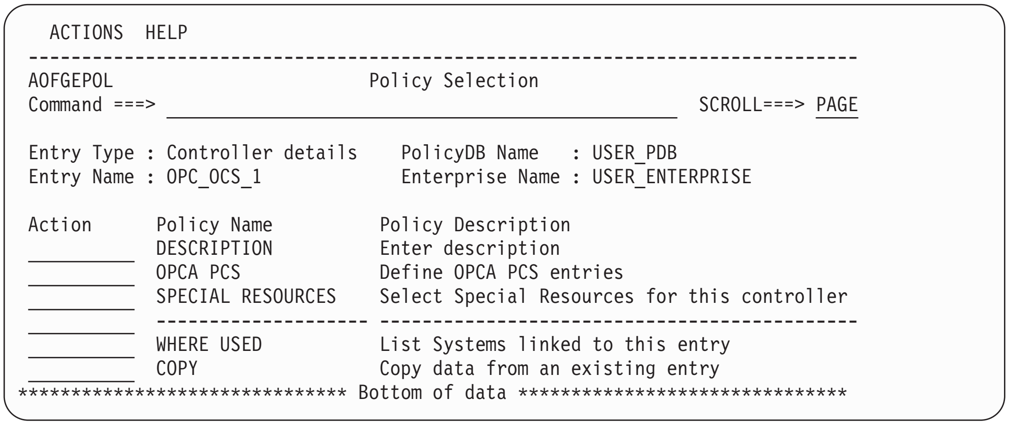
before TWS Automation will not reset operations that ended in error while it was down. The NetView interface consists of both the NetView SA z/OS TWS Automation address space and its associated NetView SSI address space, over which requests flow from TWS to TWS Automation. When the interface is down, TWS Automation sets any operations destined for a NetView workstation to error status with an error code of UNTV. If no value is coded, no operations will be reset when the interface becomes available again. The default value is NEVER .

For more information refer to *IBM Tivoli System Automation for z/OS TWS Automation Programmer's Reference and Operator's Guide*.

OPC Controller Details Policy Object

Figure 195 shows the policy items that are displayed when you are creating a new OPC controller detail or when you select an existing OPC controller detail from the related Entry Name Selection panel.

Note that if multiple controllers are running on the same system they must share the same Controller Details (OCS) definitions. If more than one OCS policy object is linked to the same system, the last is used.



```
  ACTIONS  HELP
 ------------------------------------------------------------------------------
 AOFGEPOL                       Policy Selection
 Command ===> ________________________________________________  SCROLL===> PAGE

 Entry Type : Controller details    PolicyDB Name   : USER_PDB
 Entry Name : OPC_OCS_1             Enterprise Name : USER_ENTERPRISE

 Action      Policy Name          Policy Description
 __________  DESCRIPTION          Enter description
 __________  OPCA PCS             Define OPCA PCS entries
 __________  SPECIAL RESOURCES    Select Special Resources for this controller
 __________  -------------------- ---------------------------------------------
 __________  WHERE USED           List Systems linked to this entry
 __________  COPY                 Copy data from an existing entry
****************************** Bottom of data *******************************
```

Figure 195. Policy Selection Panel for OPC Controller Details

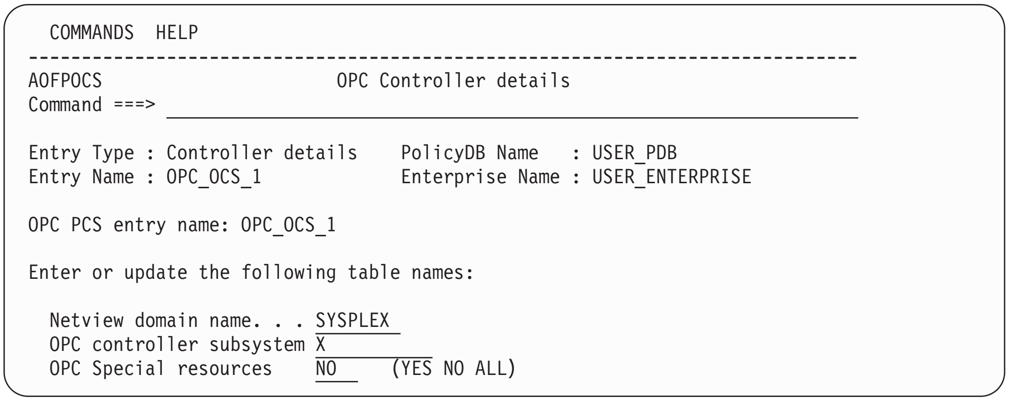
The unique policy items for OPC controller details are:

OPCA PCS
With this policy item you either specify the NetView domain that the OPC controller resides on or specify that the local sysplex is to be searched for the active controller when required. You must also specify the subsystem name for the OPC controller. For more information, refer to "OPCA PCS Policy Item" on page 267.

SPECIAL RESOURCES
Use this policy item to select one or more special resources as defined with the (OSR) Special Resources policy object for the current controller. These selected special resources for TWS are used on the systems that the current controller details entry is linked to. (You can view the systems that this entry is linked to, and link or unlink this entry to systems using the WHERE USED policy item).

OPCA PCS Policy Item

The panel shown in Figure 196 appears if you select the OPCA PCS policy item from the Policy Selection panel for TWS controller details.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPOCS                      OPC Controller details
Command ===> _________________________________________________________________

Entry Type : Controller details    PolicyDB Name   : USER_PDB
Entry Name : OPC_OCS_1             Enterprise Name : USER_ENTERPRISE

OPC PCS entry name: OPC_OCS_1

Enter or update the following table names:

  Netview domain name. . . SYSPLEX_
  OPC controller subsystem X_______
  OPC Special resources    NO__   (YES NO ALL)
```

Figure 196. OPC Controller Details Panel

The OPCA PCS automation definition defines the NetView domain that the TWS Controller resides on and specifies the MVS subsystem name for the TWS Controller system.

Fill in the following fields:

NetView domain name
Specify the parameter as either a NetView domain ID (up to five characters) or the keyword SYSPLEX. If a domain ID is specified, it indicates that the TWS Controller always runs on the specified domain. The keyword SYSPLEX indicates that the TWS Controller may be run on any one of the systems in the local sysplex.

OPC controller subsystem
The MVS subsystem name of the TWS Controller as defined in the IEFSSN*xx* member.

Note:
When defining the TWS Controller to SA z/OS, this name must be used as the subsystem name and job name.

OPC Special resources
The following entries are valid:

YES The OPC special resources will be set for any SA z/OS-monitored subsystem that is specified in the OPC special resource definitions panel (see Figure 198 on page 269). Ensure that OPC Special Resources are defined if using this option.

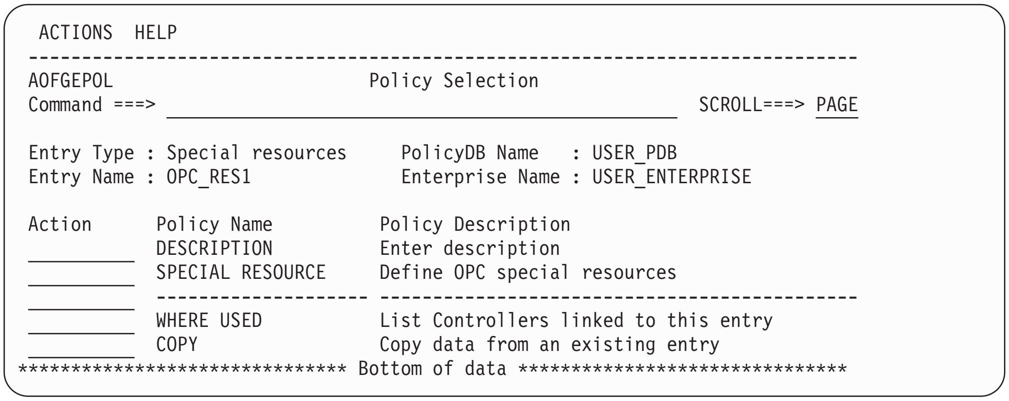
Note:
If no OPC Special Resources were defined, and the OPC Special Resource SRSTAT=YES is defined in the OPC Controller details panel, the value is automatically reset to NO during configuration build

NO No Special Resources will be set. This is the default.

ALL Special Resources will be set for all SA z/OS Monitored subsystems. No Special Resource entries are required.

OPC Special Resources Policy Object

Figure 197 shows the policy items that are displayed when you select an OPC special resource from the related Entry Name Selection panel.



```
 ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Special resources     PolicyDB Name   : USER_PDB
Entry Name : OPC_RES1              Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  SPECIAL RESOURCE     Define OPC special resources
__________  -------------------- ---------------------------------------------
__________  WHERE USED           List Controllers linked to this entry
__________  COPY                 Copy data from an existing entry
****************************** Bottom of data *******************************
```

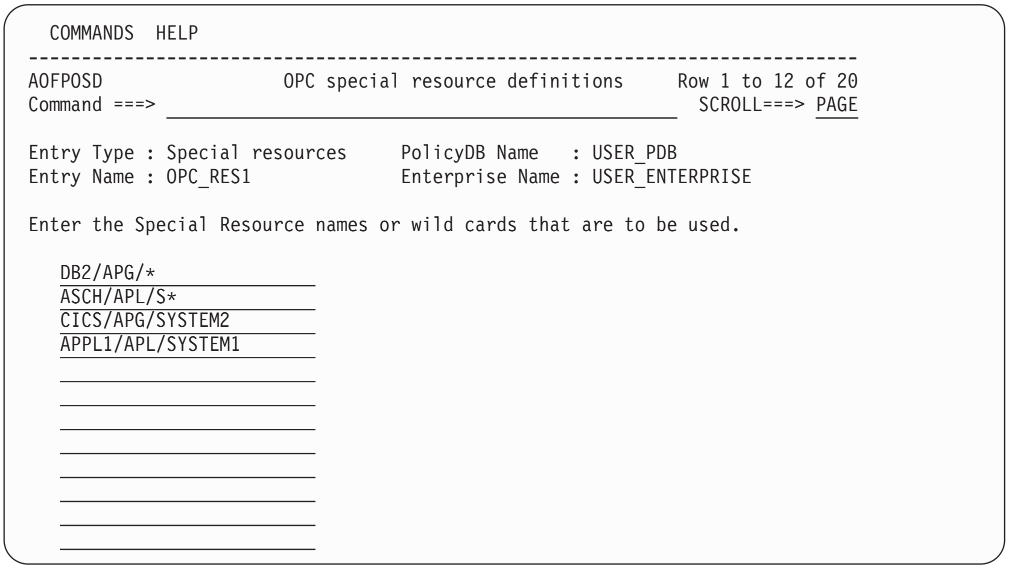
Figure 197. Policy Selection Panel for OPC Special Resources

The unique policy item for an OPC special resource is:

SPECIAL RESOURCE
With this policy item you can specify resources as special resources for TWS. SA z/OS will display the OPC special resource definitions panel. For a detailed description of this panel, refer to “SPECIAL RESOURCE Policy Item.”

SPECIAL RESOURCE Policy Item

With this policy item SA z/OS is able to globally control the creation and setting of OPC Special resources based on the status of SA z/OS monitored subsystems. This allows TWS to resolve job scheduling dependencies based on the status of SA z/OS automated resources. A group or groups of special resources can be linked to a controller definition defined with option 31 (see “Defining Automation for TWS Components” on page 260).



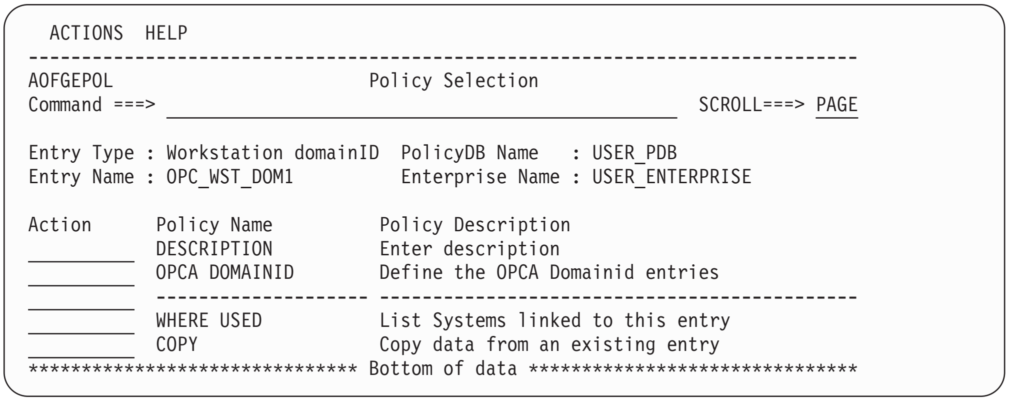
```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPOSD                 OPC special resource definitions     Row 1 to 12 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Special resources     PolicyDB Name   : USER_PDB
Entry Name : OPC_RES1              Enterprise Name : USER_ENTERPRISE

Enter the Special Resource names or wild cards that are to be used.

   DB2/APG/*_______________
   ASCH/APL/S*_____________
   CICS/APG/SYSTEM2________
   APPL1/APL/SYSTEM1_______
   ________________________
   ________________________
   ________________________
   ________________________
   ________________________
   ________________________
   ________________________
   ________________________
```

Figure 198. OPC Special Resource Definitions Panel

Fill in your special resource names with up to 24 characters. Wildcard characters and patterns may be entered.

OPC Workstation Domains Policy Object

Figure 199 shows the policy items that are displayed when you select an OPC workstation domain from the related Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Workstation domainID  PolicyDB Name   : USER_PDB
Entry Name : OPC_WST_DOM1          Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  OPCA DOMAINID        Define the OPCA Domainid entries
__________  -------------------- --------------------------------------------
__________  WHERE USED           List Systems linked to this entry
__________  COPY                 Copy data from an existing entry
****************************** Bottom of data *******************************
```

Figure 199. Policy Selection Panel for OPC Workstation Domain

The unique policy item for an OPC workstation domain is:

OPCA DOMAINID

If you select this policy item, it leads to the Code Processing panel where you specify how SA z/OS should react if TWS issues the specified message.

For a detailed description of this panel and how to define code processing policy on it, refer to “Code Processing for OPC Workstation Domains” on page 270.

For more information you should also refer to *IBM Tivoli System Automation for z/OS TWS Automation Programmer’s Reference and Operator’s Guide* under OPCA DOMAINID.

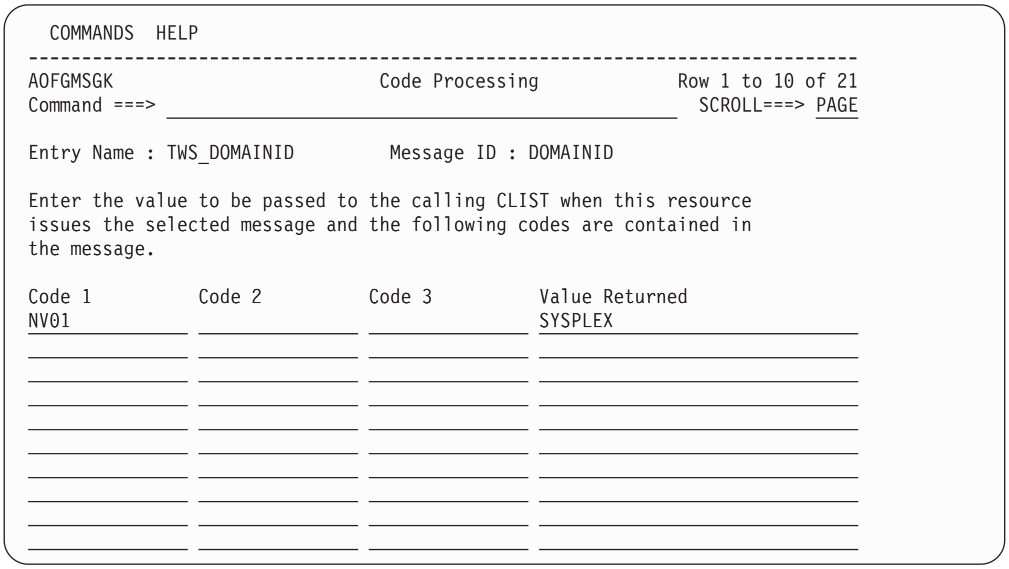
Code Processing for OPC Workstation Domains

You can use this policy item to define the system or sysplex that SA z/OS routes the request that is associated with the message to. (See Figure 200.)

Use the entry in the field **Code 1**, NV*xx*, to specify a NetView automatic general workstation with a NetView domain index of *xx*. This is resolved in the Controller NetView into the target NetView system through the definitions in the **Value Returned** field. These can be:

- The NetView domain ID
- The name of a remote sysplex. SA z/OS determines one of the systems in the remote sysplex that has a Gateway connection with the local system. This system is then used to route the command to.

  Note: A sysplex name should only be used for commands that operate sysplex-wide.
- SYSPLEX. All SA z/OS systems in the local sysplex are queried for the status of the application that is associated with the job name of the request.
- *LOCAL. The request is processed on the same system that the TWS controller runs on.

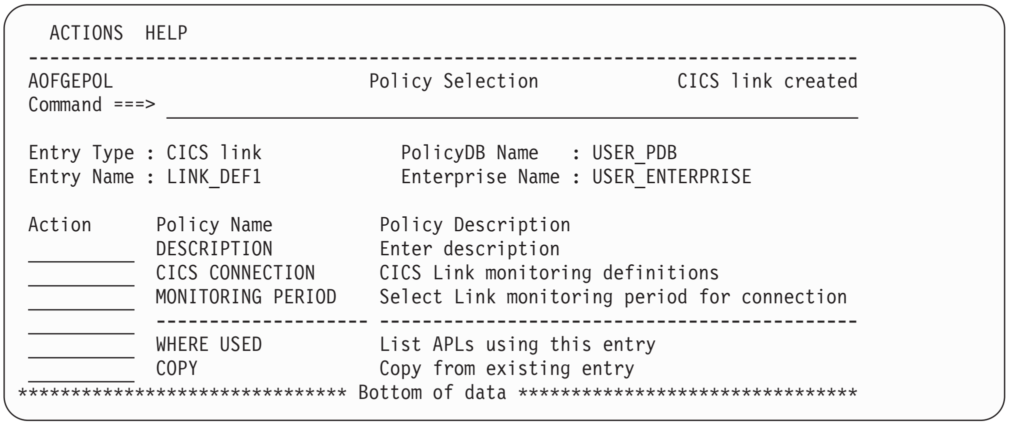


```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMSGK                        Code Processing           Row 1 to 10 of 21
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Name : TWS_DOMAINID        Message ID : DOMAINID

Enter the value to be passed to the calling CLIST when this resource
issues the selected message and the following codes are contained in
the message.

Code 1          Code 2          Code 3          Value Returned
NV01___________ _______________ _______________ SYSPLEX______________________
_______________ _______________ _______________ _____________________________
_______________ _______________ _______________ _____________________________
_______________ _______________ _______________ _____________________________
_______________ _______________ _______________ _____________________________
_______________ _______________ _______________ _____________________________
_______________ _______________ _______________ _____________________________
_______________ _______________ _______________ _____________________________
_______________ _______________ _______________ _____________________________
_______________ _______________ _______________ _____________________________
```

Figure 200. Code Processing Panel for TWS Workstation Domain IDs

CICS Link Definitions Policy Object

Figure 201 on page 271 shows the policy items that are displayed when you create or modify a link definition for CICS from the related Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection            CICS link created
Command ===> _________________________________________________________________

Entry Type : CICS link             PolicyDB Name   : USER_PDB
Entry Name : LINK_DEF1             Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  CICS CONNECTION      CICS Link monitoring definitions
__________  MONITORING PERIOD    Select Link monitoring period for connection
__________  -------------------- ---------------------------------------------
__________  WHERE USED           List APLs using this entry
__________  COPY                 Copy from existing entry
******************************* Bottom of data ********************************
```

Figure 201. Policy Selection Panel for CICS Link Definitions

The unique policy items for CICS link definitions are:

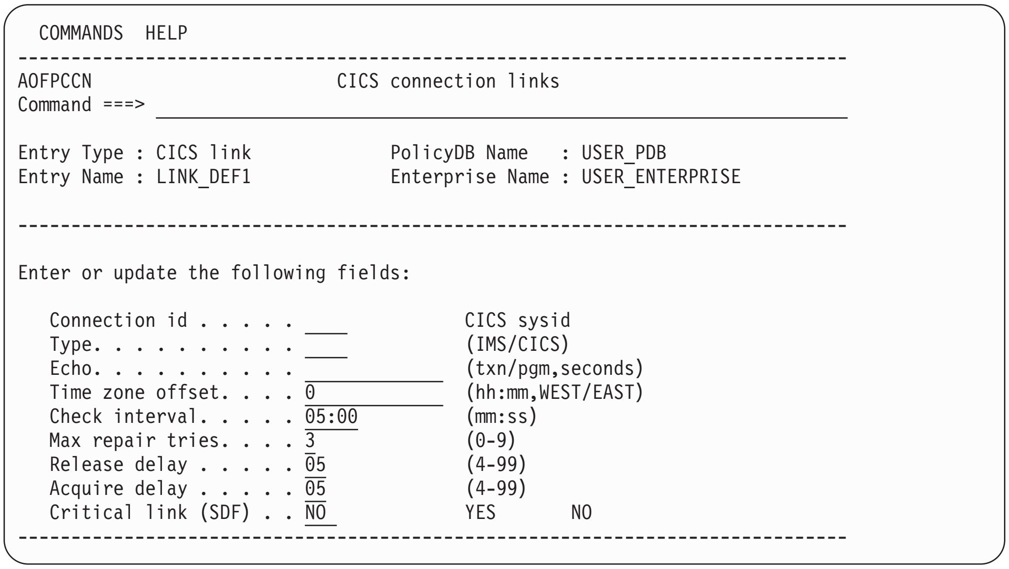
CICS CONNECTION
This policy item lets you describe the interregion and intersystem communication (IRC and ISC) links so that they can be monitored by SA z/OS CICS Automation. For more information on how to do this, see “CICS CONNECTION Policy Item.” Also refer to *IBM Tivoli System Automation for z/OS CICS Automation Programmer’s Reference and Operator’s Guide*, section CONNECTION.

MONITORING PERIOD
With this policy item you can select one monitoring period which has previously been defined with option *(CVP) Link Monitor periods* for the current CICS link definition (CICS connection).

For more information on link definitions, refer to *IBM Tivoli System Automation for z/OS CICS Automation Programmer’s Reference and Operator’s Guide*.

CICS CONNECTION Policy Item

If you select policy item CICS CONNECTION from the Policy Selection panel for CICS links, SA z/OS displays the following panel where you describe the inter-region and inter-system communication (IRC and ISC) links to be monitored by SA z/OS CICS Automation.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPCCN                       CICS connection links
Command ===> _________________________________________________________________

Entry Type : CICS link            PolicyDB Name   : USER_PDB
Entry Name : LINK_DEF1            Enterprise Name : USER_ENTERPRISE

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

Enter or update the following fields:

   Connection id . . . . . ____            CICS sysid
   Type. . . . . . . . . . ____            (IMS/CICS)
   Echo. . . . . . . . . . _____________   (txn/pgm,seconds)
   Time zone offset. . . . 0____________   (hh:mm,WEST/EAST)
   Check interval. . . . . 05:00           (mm:ss)
   Max repair tries. . . . 3               (0-9)
   Release delay . . . . . 05              (4-99)
   Acquire delay . . . . . 05              (4-99)
   Critical link (SDF) . . NO_             YES       NO
------------------------------------------------------------------------------
```

Figure 202. CICS Connection® Links Panel

Fill in the following fields:

Connection id
The 4 character symbolic name by which this link is known to this CICS subsystem. This is defined on the CICS subsystem itself (SYSID).

Type Indicates the type of subsystem this link is attached to, valid values are ″CICS″ or ″IMS″, this is used only when you want echoplexing to occur.

Echo depends on the entry in the *Type* field:

CICS The transaction name used to call the echoplex program on the target subsystem.

IMS Any non-blank character.

The time parameter is used to define the time delay in seconds after which an echo response must have been received from the remote system before a message is issued. Valid values are from 1 to 99, default 3.

Time zone offset
This parameter specifies the remote systems time offset relative to the UTC (Greenwich mean time). This is applied to the times specified in the DAY keyword to determine when monitoring is active for the remote system. Default value is 0.

Check interval
The time interval between 2 consecutive checks of a connection. Valid values are from 00:05 to 99:59, the default value is 05:00.

Max repair tries
Specifies the number of automatic actions that are to take place after the detection of a connection failure. Valid values are 0 to 9, default value is 3.

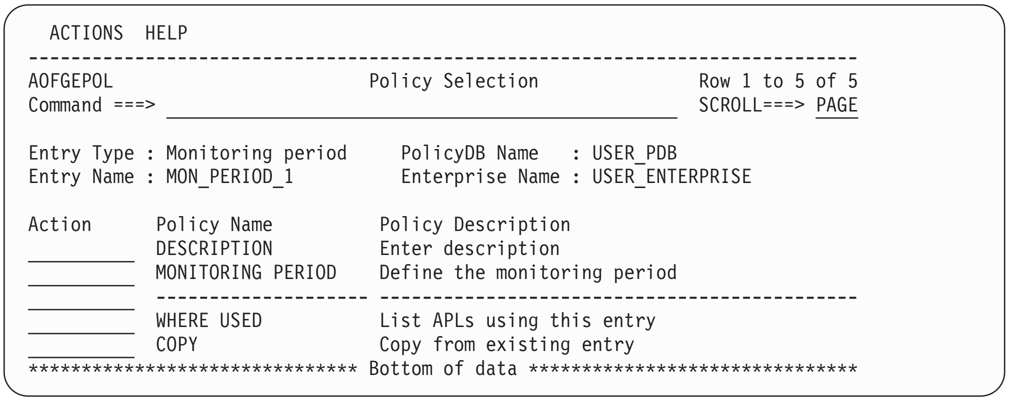
Release delay
Specifies the time delay in seconds needed by the system to release a connection. Valid values are 4 to 99, default value is 5.

Acquire delay
Specifies the time delay in seconds needed by the system to acquire a connection. Valid values are 4 to 99, default value is 5.

Critical link
Specifies whether a link failure for this link will change the severity shown on the SDF panels, default value is NO.

CICS Link Monitor Periods Policy Object

Figure 203 shows the policy items that are displayed when you select a monitoring period CICS from the related Entry Name Selection panel or if you are creating a new monitoring period.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection             Row 1 to 5 of 5
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Monitoring period     PolicyDB Name   : USER_PDB
Entry Name : MON_PERIOD_1          Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  MONITORING PERIOD    Define the monitoring period
__________  -------------------- ----------------------------------------------
__________  WHERE USED           List APLs using this entry
__________  COPY                 Copy from existing entry
****************************** Bottom of data *******************************
```

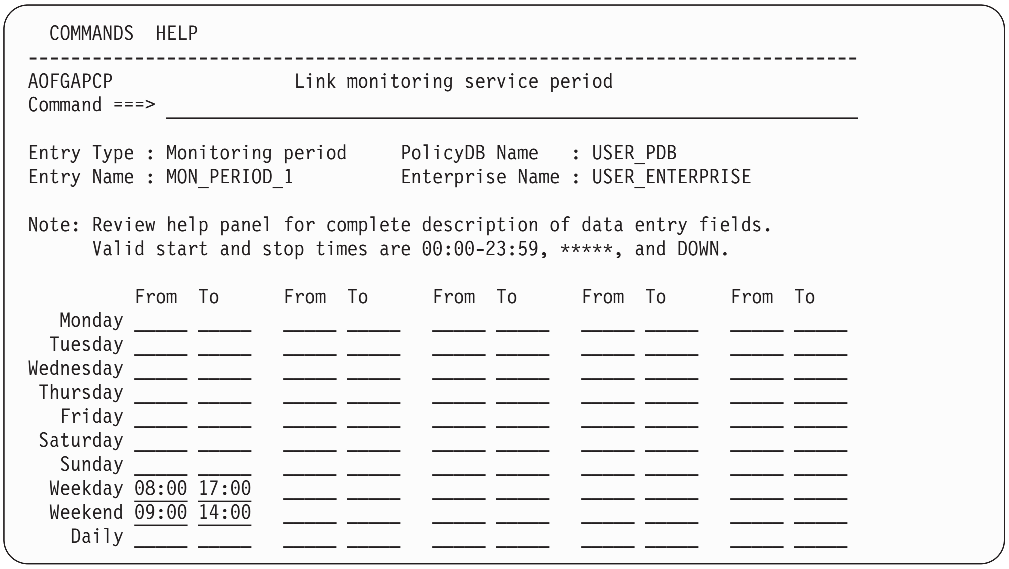
Figure 203. Policy Selection Panel for CICS Link Monitor Periods

The unique policy item for a CICS monitoring period is:

MONITORING PERIOD
With this policy item you can specify time periods when the current CICS connection should be monitored. You specify these time periods as described in “MONITORING PERIOD Policy Item.”

MONITORING PERIOD Policy Item

If you select policy item MONITORING PERIOD from the Policy Selection panel for CICS monitoring periods, SA z/OS displays the following panel where you define the time frames for when the CICS connection link related to the *(CCN) Link definition* should be active.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGAPCP                 Link monitoring service period
Command ===> _________________________________________________________________

Entry Type : Monitoring period     PolicyDB Name   : USER_PDB
Entry Name : MON_PERIOD_1          Enterprise Name : USER_ENTERPRISE

Note: Review help panel for complete description of data entry fields.
      Valid start and stop times are 00:00-23:59, *****, and DOWN.

          From  To      From  To      From  To      From  To      From  To
   Monday _____ _____   _____ _____   _____ _____   _____ _____   _____ _____
  Tuesday _____ _____   _____ _____   _____ _____   _____ _____   _____ _____
Wednesday _____ _____   _____ _____   _____ _____   _____ _____   _____ _____
 Thursday _____ _____   _____ _____   _____ _____   _____ _____   _____ _____
   Friday _____ _____   _____ _____   _____ _____   _____ _____   _____ _____
 Saturday _____ _____   _____ _____   _____ _____   _____ _____   _____ _____
   Sunday _____ _____   _____ _____   _____ _____   _____ _____   _____ _____
  Weekday 08:00 17:00   _____ _____   _____ _____   _____ _____   _____ _____
  Weekend 09:00 14:00   _____ _____   _____ _____   _____ _____   _____ _____
    Daily _____ _____   _____ _____   _____ _____   _____ _____   _____ _____
```

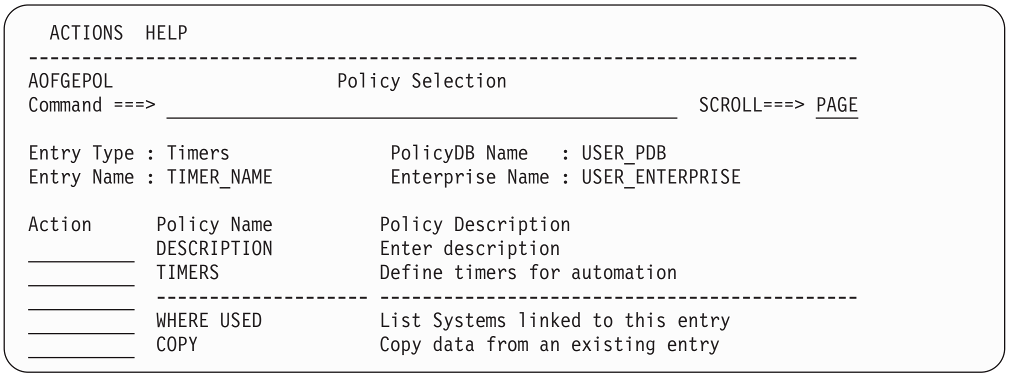
Figure 204. Link Monitoring Service Period Panel

For information about valid values and the syntax of the time frames, refer to the online help.

Timers Policy Object

The Timers policy objects allow you to define sets of timers that run commands or automation procedures at scheduled times.

Figure 205 shows the policy items that are displayed when a new Timers entry is defined or when an existing Timers entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                     Policy Selection
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Timers               PolicyDB Name   : USER_PDB
Entry Name : TIMER_NAME           Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  TIMERS               Define timers for automation
__________  -------------------- --------------------------------------------
__________  WHERE USED           List Systems linked to this entry
__________  COPY                 Copy data from an existing entry
```

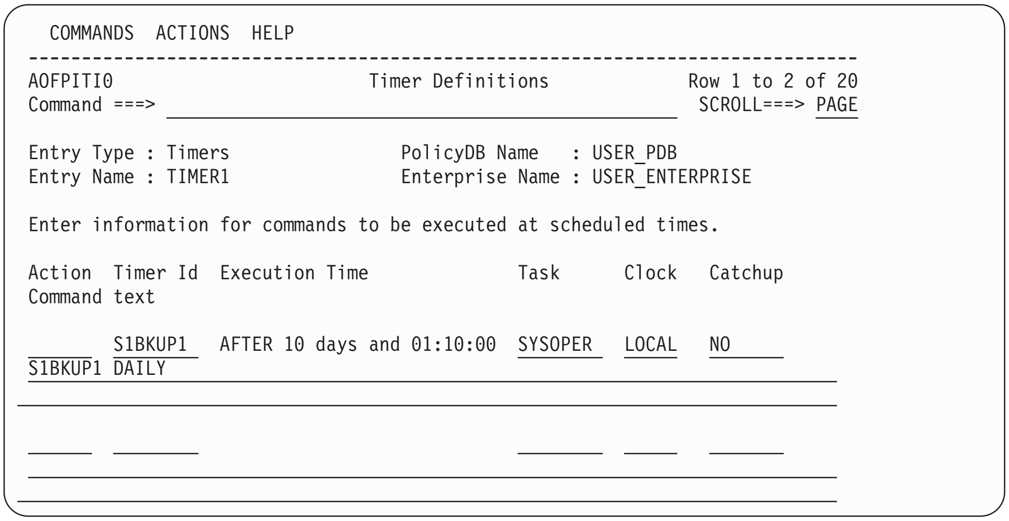
Figure 205. Policy Selection Panel for Timers

The unique policy item for Timers is:

TIMERS
You define a timer in the Timer specification panel shown in Figure 206 on page 275.

TIMERS Policy Item

SA z/OS displays the following panel if you select the TIMERS policy item from the Policy Selection panel for Timers.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFPITI0                        Timer Definitions            Row 1 to 2 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Timers                PolicyDB Name   : USER_PDB
Entry Name : TIMER1                Enterprise Name : USER_ENTERPRISE

Enter information for commands to be executed at scheduled times.

Action  Timer Id  Execution Time              Task      Clock   Catchup
Command text

______  S1BKUP1   AFTER 10 days and 01:10:00  SYSOPER   LOCAL   NO
S1BKUP1 DAILY
_____________________________________________________________________________

______  ________                              ________  _____   _______
____________________________________________________________________________
_____________________________________________________________________________
```

Figure 206. Timer Definitions Panel

You should note that, at run time, the commands that you define here are converted to the NetView CHRON format.

A timer definition consists of:

Timer ID
When a user timer is defined with the SAVE-option outside the control of SA z/OS, it is recommended that a unique timer ID is specified to avoid duplicate timers being created at automation control file (ACF) reload.

Task
Specify the automated function or NetView task name where the command should run. You may also specify a NetView operator group by using a '+' sign as the first character (for example, +OPERGRP). To run under the primary programmed task (PPT) specify PPT.

Note: Not all commands can run under the PPT.

Clock
Specifies whether the command is to run on a Greenwich Mean Time schedule or the local system time. You can enter either GMT or LOCAL.

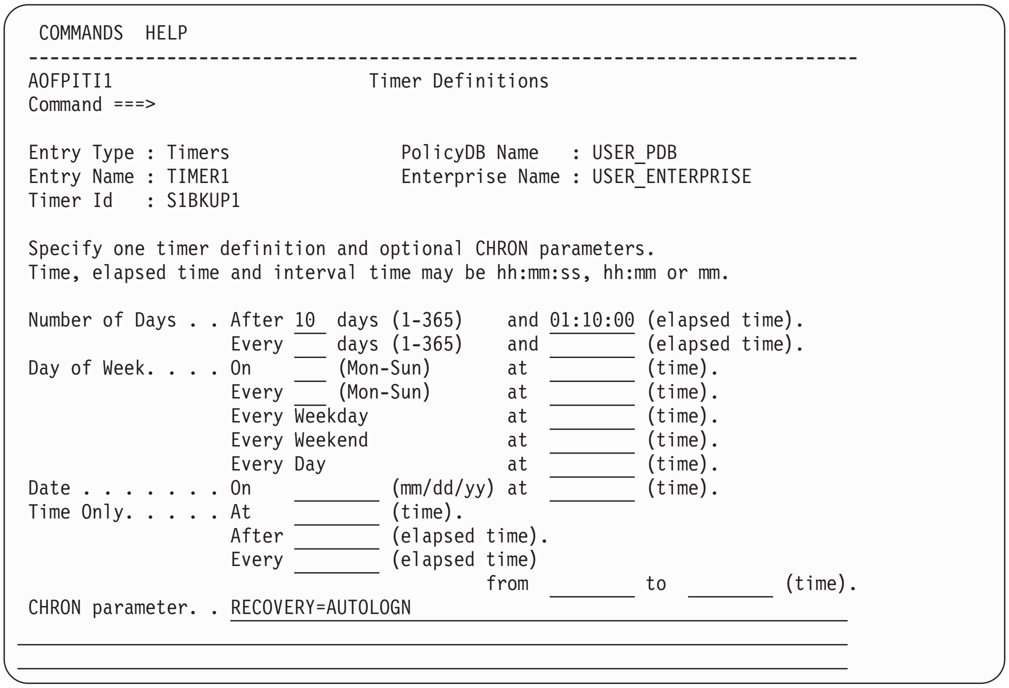
Catchup
Specify whether a saved timer should be caught up (that is, it should still be run) after a system outage.

Note: This parameter will not work for every timer.

Command text
The actual timer-driven command or automation procedure that SA z/OS issues or calls.

The **Execution Time** field cannot be filled in from this panel, but it shows the time at which the timer has been scheduled. This is the result of completing the Timer Definition panel (selected with an S in the **Action** column).

Use the S command code to define when the timer executes. This leads to the second Timer Definitions panel, as shown in Figure 207 on page 276.



```
 COMMANDS  HELP
------------------------------------------------------------------------------
AOFPITI1                        Timer Definitions
Command ===>

Entry Type : Timers                PolicyDB Name   : USER_PDB
Entry Name : TIMER1                Enterprise Name : USER_ENTERPRISE
Timer Id   : S1BKUP1

Specify one timer definition and optional CHRON parameters.
Time, elapsed time and interval time may be hh:mm:ss, hh:mm or mm.

Number of Days . . After 10  days (1-365)    and 01:10:00 (elapsed time).
                   Every ___ days (1-365)    and ________ (elapsed time).
Day of Week. . . . On    ___ (Mon-Sun)       at  ________ (time).
                   Every ___ (Mon-Sun)       at  ________ (time).
                   Every Weekday             at  ________ (time).
                   Every Weekend             at  ________ (time).
                   Every Day                 at  ________ (time).
Date . . . . . . . On    ________ (mm/dd/yy) at  ________ (time).
Time Only. . . . . At    ________ (time).
                   After ________ (elapsed time).
                   Every ________ (elapsed time)
                                           from  ________ to  ________ (time).
CHRON parameter. . RECOVERY=AUTOLOGN_________________________________________
______________________________________________________________________________
______________________________________________________________________________
```

Figure 207. Timer Definitions Panel 2

You can use only *one* of the available options to define a timer, and additionally any valid NetView CHRON timer parameters.Note that entering the colons is optional for all the time fields.

The time and elapsed time can be entered as:

1. Hours, minutes and seconds
2. Hours and minutes
3. Minutes only (hours then defaults to 00)

Valid values are 00-23 (hours) and 00-59 (minutes and seconds). Specifying 24:00:00 is also valid for elapsed time.

You can define timers in the various sections of the panel, based on the time interval, as follows:

Time interval	When the timer should expire	
Number of Days, based on the elapsed time.	**After**	After a certain number of days and specified time interval.
	Every	Periodically, after a specified number of days and specified time interval.

Time interval	When the timer should expire
Day of Week, based on the day of the week.	**On** Only once at a certain time within the next seven days. **Every** Once a week at a certain time. **Every Weekday** At the same time every weekday (Monday through Friday). **Every Weekend** At the same time every weekend (Saturday or Sunday). **Every Day** At the same time every day.
Date, if the timer is to expire on a particular day.	**On** On a particular day. **Note:** Only future dates are allowed. If the year is 20 or less it is assumed to be 20*xx*.
Time Only, based on a specified time.	**At** At a certain time within the next 24 hours. **After** A certain period of time after the timer has been set. **Every** At regular intervals. Optionally you may specify a time frame (using the **from** and **end** fields) where the timer expires at the regular interval.

The CHRON parameter can be any combination of NetView CHRON timer parameters, such as RECOVERY, NOTIFY, etc. However, the following parameters must *not* be specified because they are automatically created from the definitions that you make on this panel and the preceding panel:

AT	COMMAND	GMT/LOCAL	ROUTE
AFTER	EVERY	ID	SAVE/NOSAVE

You can find more details about the NetView CHRON commands that SA z/OS generates during initialization from the online help.

Timeout Settings Policy Object

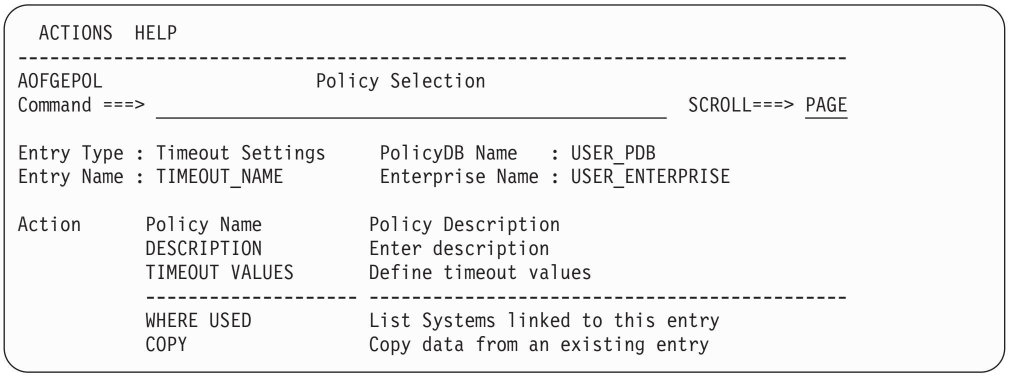
This policy object enables you to create sets of timeout definitions. Each timeout definition can specify timeout values for:

- System, NetView, or SA z/OS commands
- Cross-domain logon requests
- Your own timeout variables

If the time limit is exceeded, it is assumed that an error condition exists and an appropriate action is taken.

Although only two specific fields, Wait and XDOM, are given, you can add other timeout names that may be unique to your environment. Figure 208 on page 278 shows the policy items that are displayed when a new Timeout Setting entry is defined or when an existing Timeout Setting entry is selected from the Entry Name

Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                    Policy Selection
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Timeout Settings     PolicyDB Name   : USER_PDB
Entry Name : TIMEOUT_NAME         Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            DESCRIPTION          Enter description
            TIMEOUT VALUES       Define timeout values
            -------------------- ---------------------------------------------
            WHERE USED           List Systems linked to this entry
            COPY                 Copy data from an existing entry
```

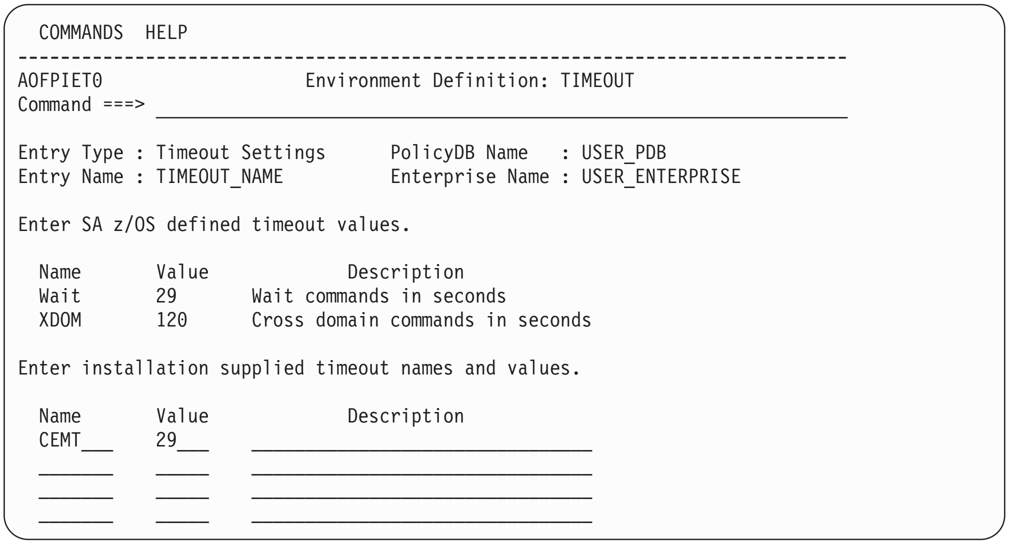
Figure 208. Policy Selection Panel for Timeout Settings

The unique policy item for Timeout Settings is:

TIMEOUT VALUES
SA z/OS defines two timeout variables, XDOMTIME and WAITTIME. Values for these variables can be set in the Wait and XDOM Value fields of the Environment Definition: TIMEOUT panel.

TIMEOUT VALUES Policy Item

Figure 209 shows the definition panel for a Timeout Settings.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIET0                   Environment Definition: TIMEOUT
Command ===> _________________________________________________________________

Entry Type : Timeout Settings      PolicyDB Name   : USER_PDB
Entry Name : TIMEOUT_NAME          Enterprise Name : USER_ENTERPRISE

Enter SA z/OS defined timeout values.

  Name       Value            Description
  Wait       29      Wait commands in seconds
  XDOM       120     Cross domain commands in seconds

Enter installation supplied timeout names and values.

  Name       Value            Description
  CEMT___    29___   ________________________________
  _______    _____   ________________________________
  _______    _____   ________________________________
  _______    _____   ________________________________
```

Figure 209. Environment Definition: TIMEOUT Panel

The following timeout values can be entered:

Wait
Defines how long automation processing waits for a response after issuing a system, NetView, or SA z/OS system operations command. The wait parameter is used in the automation code to determine when a response error condition has occurred. Depending on which part of the code was being used at the time of the error and how critical the command is, an action is taken. The default time is 29 seconds. The range of values is 1 to 32000 seconds.

It is recommended that you choose a value less than 30 seconds to avoid the NetView message DSI594A being displayed in SA z/OS full screen function. The field, wait, creates the global variable WAITTIME.

XDOM

Defines the maximum amount of time that message forwarding routines wait for a response from a cross-domain logon command. If this value is exceeded, the routines assume that the logon attempt failed. The default time is 120 seconds. The range of values is 1 to 32000 seconds. XDOM creates the global variable XDOMTIME.

Installation supplied timeout names and values

This section is used to define your own timeout values. Common global variables are created containing these values, and these variables can be used in your own automation procedures. The value of these variables must be in the range of 1 to 32000 seconds.

Note:
You must ensure that the names of any global variables you create do not clash with SA z/OS external or internal global variable names. SA z/OS external global variables are documented in *IBM Tivoli System Automation for z/OS Customizing and Programming*. In addition, you should not use names beginning with AOF, EHK, EVE, EVI, EVJ, ING, ISQ, or SUBS.

For example, you can define the name TEST with a value of 30. This creates a global variable called TESTTIME containing the value 30. The variable TESTTIME can now be used in any automation procedure. You can enter an optional description of the timeout variable for reference.

Note:
The TESTTIME variable is initialized after SA z/OS has initialized its own environment from the automation control file. This means you should not use a variable such as TESTTIME in a routine which is invoked before SA z/OS is fully initialized.

Tape Attendance Policy Object

This policy object allows you to specify whether you want to monitor tape mounts on this system. If you do want to monitor tape mounts, you must enter warning and alert thresholds (in minutes and seconds) indicating how long a wait is allowed in mounting a tape before the response is considered unsatisfactory.

You can customize SA z/OS to respond to the IEF238D message while tape drives are attended by operators. This can be achieved by specifying an entry with code definitions for the IEF238D message in the MESSAGES/USER DATA policy item of the MVS Component policy object.

The codes that are passed when checking for a code match with this entry are one of the following, depending on which of these terms is included in the IEF238D message:

- The job name as the first parameter, and either DEVICE or WAIT as the second parameter

- The job name as the first parameter, and DEVICE and WAIT as the second and third parameters

Therefore the coded entries should be entered as follows:

CODE1
This should contain a pattern match for the job name that needs the response to the IEF238D message. You can specify a leading or trailing asterisk (*) to allow for generic job name matching.

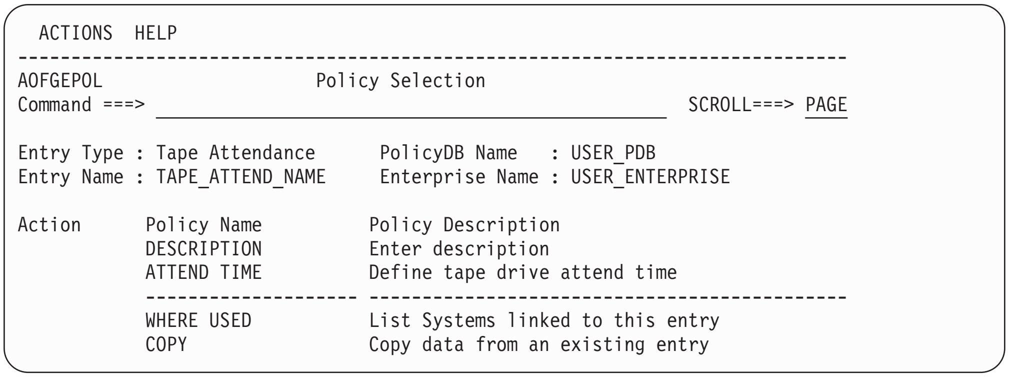
CODE2
This should contain DEVICE or WAIT or an asterisk (*) to allow for both terms.

CODE3
This should contain WAIT or an asterisk (*), or should be left blank.

Value Returned
This should contain a selection name that has a reply defined for the IEF238D message, depending on the results of the code matching process.

SA z/OS provides a NetView automation table entry that extracts the values from the message that are to be checked for a code match and passes them as CODE*x* parameters to the ISSUEACT generic routine. ISSUEACT uses the CDEMATCH common routine for the code checking and issues a reply, depending on the code match. For further information about message automation refer to the chapter "How to Add a Message to Automation" in *IBM Tivoli System Automation for z/OS Customizing and Programming*. For more details on the format of the IEF238D message, refer to the system message documentation for your particular operating system level and release.

This policy object also allows you to set a schedule for unattended tape drives. If a job requests a tape mount at a time defined as unattended, the job will be canceled. Figure 210 shows the policy items that are displayed when a new Tape Attendance entry is defined or when an existing Tape Attendance entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                    Policy Selection
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Tape Attendance      PolicyDB Name   : USER_PDB
Entry Name : TAPE_ATTEND_NAME     Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            DESCRIPTION          Enter description
            ATTEND TIME          Define tape drive attend time
            -------------------- --------------------------------------------
            WHERE USED           List Systems linked to this entry
            COPY                 Copy data from an existing entry
```

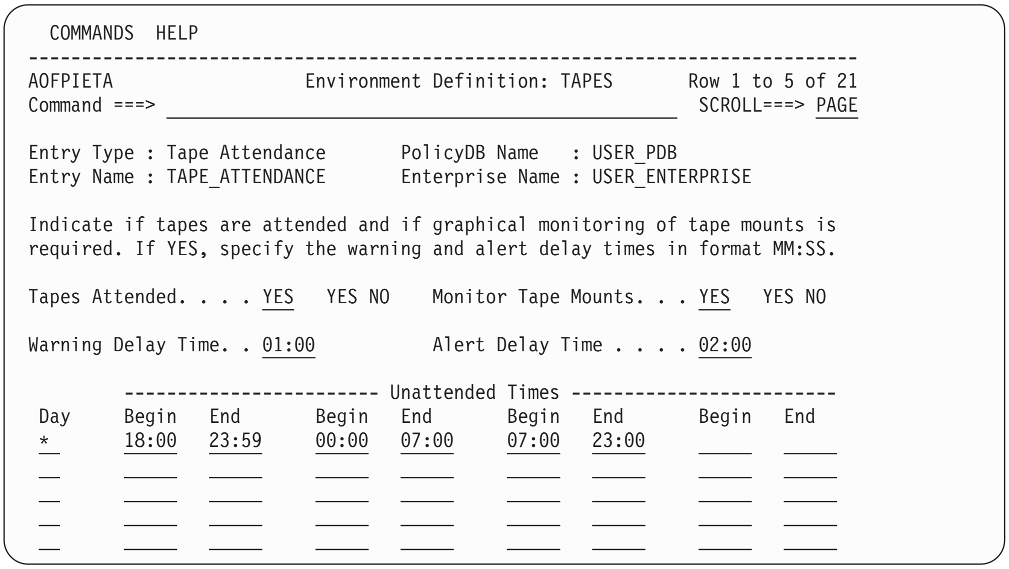
Figure 210. Policy Selection Panel for Tape Attendance

The unique policy item for Tape Attendance is:

ATTEND TIME
This policy item allows you to specify when tapes are attended and whether you want to monitor tape mounts on this system. If you do want to monitor tape mounts, you must enter warning and alert thresholds (in minutes and seconds) indicating the maximum tape mount delay that is allowed before the response is considered unsatisfactory.

ATTEND TIME Policy Item

Figure 211 shows the Environment Definition: TAPES panel.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIETA                 Environment Definition: TAPES      Row 1 to 5 of 21
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Tape Attendance       PolicyDB Name   : USER_PDB
Entry Name : TAPE_ATTENDANCE       Enterprise Name : USER_ENTERPRISE

Indicate if tapes are attended and if graphical monitoring of tape mounts is
required. If YES, specify the warning and alert delay times in format MM:SS.

Tapes Attended. . . . YES   YES NO    Monitor Tape Mounts. . . YES   YES NO

Warning Delay Time. . 01:00           Alert Delay Time . . . . 02:00

         ------------------------ Unattended Times -------------------------
 Day     Begin   End       Begin   End       Begin   End       Begin   End
 *       18:00   23:59     00:00   07:00     07:00   23:00     _____   _____
 __      _____   _____     _____   _____     _____   _____     _____   _____
 __      _____   _____     _____   _____     _____   _____     _____   _____
 __      _____   _____     _____   _____     _____   _____     _____   _____
 __      _____   _____     _____   _____     _____   _____     _____   _____
```

Figure 211. Environment Definition: TAPES Panel

The fields are:

Tapes Attended

Specify whether tape drives are attended or not. This is a global setting that indicates if there are any attended tapes attached to the system. If there are attended tapes then you can specify days and times at which they will not be attended.

Monitor Tape Mounts

Specify whether tape mounts should be monitored on the SA z/OS workstation. If you enter YES in this field, SA z/OS will automatically create a Tape Mounts object for the NMC workstation. A Tape Mounts object is an icon that represents an unsatisfied tape mount request by a job. When SA z/OS detects a tape mount request, it creates an icon under the Tape_Mount group icon. The icon is green, initially, indicating a satisfactory condition. If the tape mount request is not satisfied before the Warning interval has expired, the icon changes color to white. If the tape mount request is still not satisfied after the Alert interval has expired, the icon changes color from white to red.

The icon is deleted automatically when the requested tape is mounted or the job requesting the tape mount is canceled.

Note:

The operator can specify whether the NMC workstation view that contains the tape mount icons is automatically refreshed. It is advisable to set this to refresh automatically so that tape mounts can be monitored without manual intervention.

The name of the tape mount object is of the form

```
SSSSS.VVVVVV_DDD(D)
```

where:

SSSSS
is the NetView domain name where the mount request was issued. This may be a remote system.

VVVVVV
is the tape volume name

DDD(D)
is the device number that the tape should be mounted on. This device number is either 3 or 4 digits depending on the level of z/OS.

Warning Delay
The delay period, in minutes and seconds, after which the tape mount icon changes to the warning color of white if the tape mount has not been completed.

Alert Delay
The delay period, in minutes and seconds, between the time that the warning delay is reached and the time that the tape mount icon changes to the alert color of red, if the tape mount has not been completed.

If tapes are attended, enter days and times (24 hour clock) when unattended.
These fields are used to enter days and times when tape drives are unattended. A “*” character can be used in the Day field to specify all days of the week.

MVS Component Policy Object

This policy object allows you to apply automation to various z/OS data sets, resources and facilities, which are referred to as MVS components.

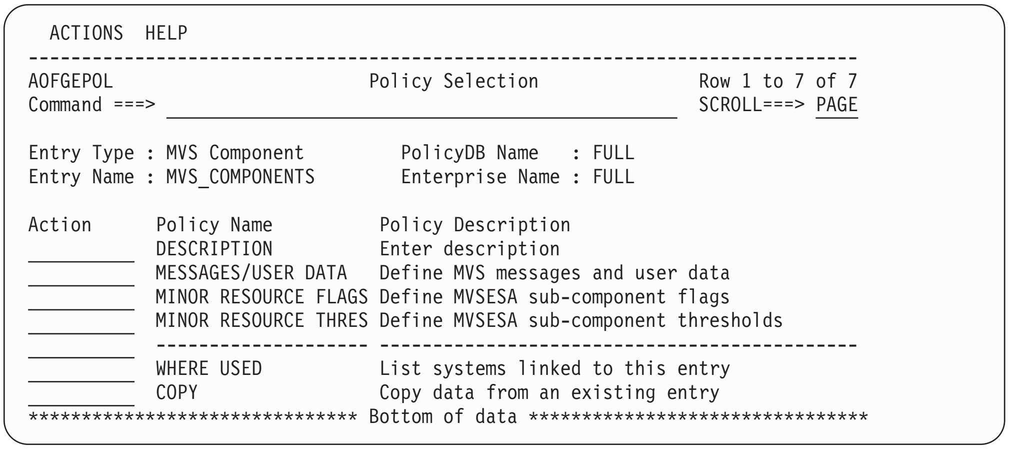
You can set error thresholds and automation flags for the following MVS components:

- MVSDUMP (SYS1.DUMP00–*xx*)
- SMFDUMP (SYS1.MAN*x*)
- LOGREC (SYS1.LOGREC)
- SYSLOG (SYS1.SYSLOG.INDX)

In addition, you can customize the recovery procedure that is used for a WTOBUF overflow and automate responses to messages that are issued for MVS components.

Policy Items for MVS Component

Figure 212 on page 283 shows the policy items that are displayed when a new MVS Component entry is selected or when an existing MVS Component entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection             Row 1 to 7 of 7
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : MVS Component         PolicyDB Name   : FULL
Entry Name : MVS_COMPONENTS        Enterprise Name : FULL

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  MESSAGES/USER DATA   Define MVS messages and user data
__________  MINOR RESOURCE FLAGS Define MVSESA sub-component flags
__________  MINOR RESOURCE THRES Define MVSESA sub-component thresholds
__________  -------------------- -------------------------------------------
__________  WHERE USED           List systems linked to this entry
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

Figure 212. Policy Selection Panel for MVS Components

The policy items for MVS Component policy objects are:

MESSAGES/USER DATA
This policy item allows you to specify automated actions, such as the issuing of a command or reply, to be taken in response to a message issued by the application. You may specify commands if the application issues messages because something went wrong with the MVSDUMP, SMFDUMP, SYSLOG or LOGREC data sets. For more information see “MESSAGES/USER DATA Policy Item.”

MINOR RESOURCE FLAGS
This policy item allows you to define flags for minor resources for the system. Minor resources are any resources, such as messages or statuses, that are specified to be subsidiary to the major resource MVSESA.

Minor resources take, by default, the automation settings of their major resources. The main purposes in defining minor resources are to:

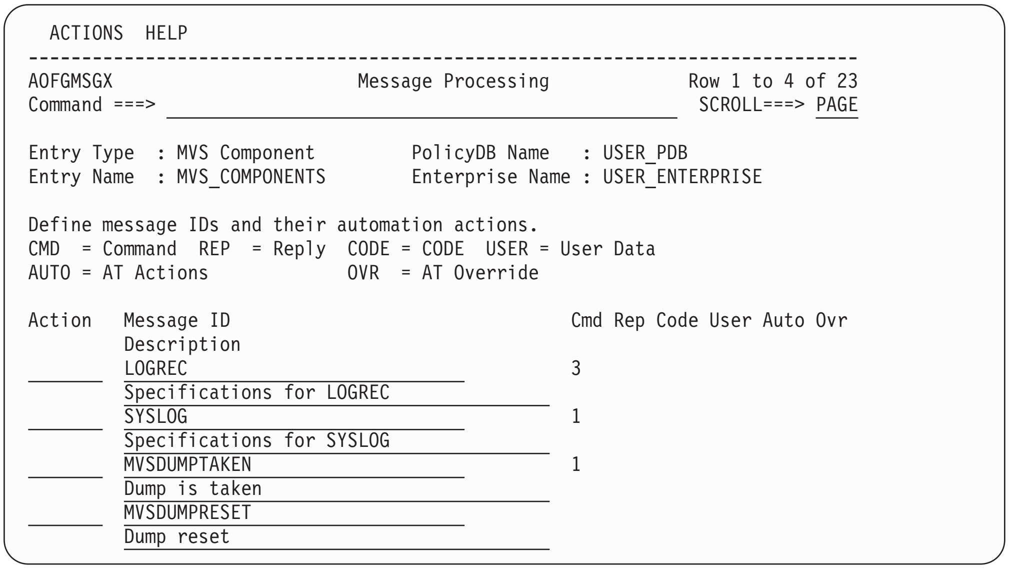
- Enable automation for minor resources
- Override the automation settings of major resources for a single minor resource, or a group of minor resources

For information on this policy item see “MINOR RESOURCE FLAGS Policy Item” on page 285. For information on the relation of minor resources and automation flags refer to “Automation Flags and Minor Resource Flags” on page 25.

MINOR RESOURCE THRES
This policy item allows you to define thresholds for minor resources for the system. For more information see “MINOR RESOURCE THRES Policy Item” on page 286 and “Thresholds for Minor Resources” on page 20.

MESSAGES/USER DATA Policy Item

If you select policy item MESSAGES/USER DATA from the Policy Selection panel for MVS Components (Figure 212), you come to the Message Processing panel shown in Figure 213 on page 284.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGMSGX                       Message Processing             Row 1 to 4 of 23
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type  : MVS Component        PolicyDB Name   : USER_PDB
Entry Name  : MVS_COMPONENTS       Enterprise Name : USER_ENTERPRISE

Define message IDs and their automation actions.
CMD  = Command  REP  = Reply  CODE = CODE  USER = User Data
AUTO = AT Actions             OVR  = AT Override

Action   Message ID                            Cmd Rep Code User Auto Ovr
         Description
_______  LOGREC                                3
         Specifications for LOGREC
_______  SYSLOG                                1
         Specifications for SYSLOG
_______  MVSDUMPTAKEN                          1
         Dump is taken
_______  MVSDUMPRESET
         Dump reset
```

Figure 213. Message Processing Panel

The MESSAGES/USER DATA policy item allows you to specify message IDs, and special ones like MVSDUMPTAKEN, MVSDUMPRESET, MVSDUMP, SMFDUMP, SYSLOG, LOGREC, and WTOBUF. The message ID must not end with an asterisk and cannot have any commas in it.

On this panel, you can define how SA z/OS should react to an issued message. Message processing provided here is the same as provided for Applications. For more information refer to "MESSAGES/USER DATA Policy Item" on page 180 and *IBM Tivoli System Automation for z/OS Customizing and Programming*

WTO Buffer Shortage Recovery

The definitions can also be used for WTO buffer shortage recovery. To define these use message ID WTOBUF and define the following values:

CODE1
Specifies the name of the job which might or might not be canceled. The job name may be specified with the wildcard character "*" as the last character.

CODE2
Must be WTO, WTOR or * to indicate which requests the job (or jobs) might or might not be canceled for. Use just * to specify WTO and WTOR requests.

CODE3
Must be blank.

Value Returned
Must be CANCEL or KEEP to indicate whether the job (or jobs) might or might not be canceled.

Example:

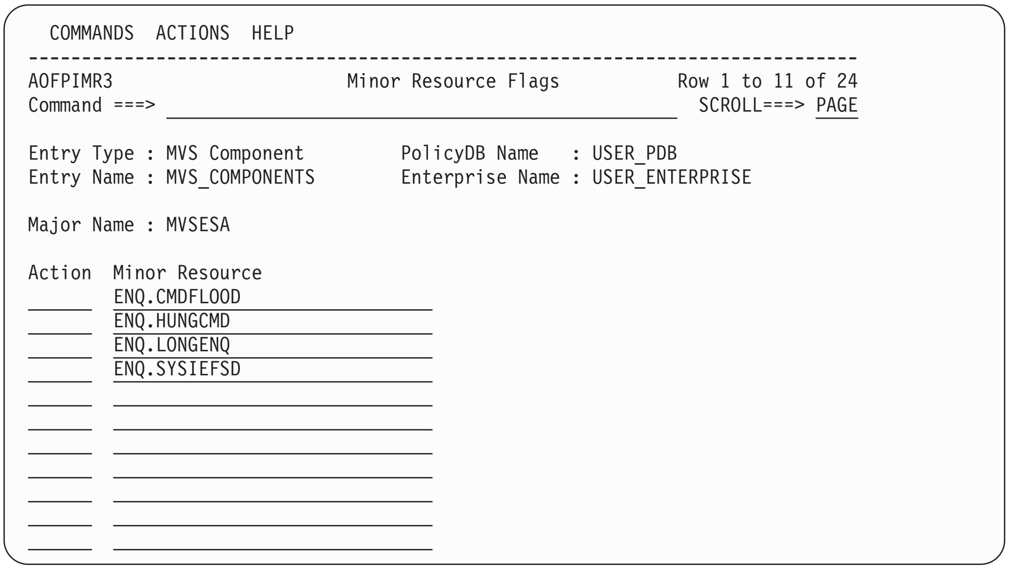
Code 1	Code 2	Code 3	Value Returned
JOB1	WTOR		KEEP
JOB2*	WTO		KEEP
JOB3*	*		CANCEL
JOB4*	*		KEEP
*	*		KEEP

To set up the default behavior for all jobs not explicitly defined, a specification of CODE1=* and CODE2=* is needed. To indicate that all other jobs might be canceled specify CANCEL in the VALUE RETURNED field, otherwise specify KEEP.

The job name *MASTER* cannot be entered in the CODE1 field. Even if your default behavior is set up to cancel all jobs that have not been explicitly defined, a cancel command is not issued against *MASTER* if it is the job name being checked. This is because *MASTER* is non-cancelable.

MINOR RESOURCE FLAGS Policy Item

The Minor Resource Flags panel, as shown in Figure 214, is displayed if you select the MINOR RESOURCE FLAGS policy item in the Policy Selection panel for the MVS Component entry type.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFPIMR3                     Minor Resource Flags           Row 1 to 11 of 24
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : MVS Component         PolicyDB Name   : USER_PDB
Entry Name : MVS_COMPONENTS        Enterprise Name : USER_ENTERPRISE

Major Name : MVSESA

Action  Minor Resource
______  ENQ.CMDFLOOD_________________
______  ENQ.HUNGCMD__________________
______  ENQ.LONGENQ__________________
______  ENQ.SYSIEFSD_________________
______  _____________________________
______  _____________________________
______  _____________________________
______  _____________________________
______  _____________________________
______  _____________________________
______  _____________________________
```

Figure 214. Minor Resource Flags Panel

This panel allows you to manage automation flags for minor resources.

Major Name
This field shows the major resource name.

Action
Use this field to select (S) a minor resource for further processing or to delete (D) a minor resource.

Minor Resource
Use this field to specify the minor resource name. The following rules apply to the minor resource names:

- The name cannot contain any of the following characters: comma, single quotation mark, double quotation mark, left or right parenthesis, equal sign or a blank.
- The name cannot start or end with a period.
- The name cannot contain consecutive periods.

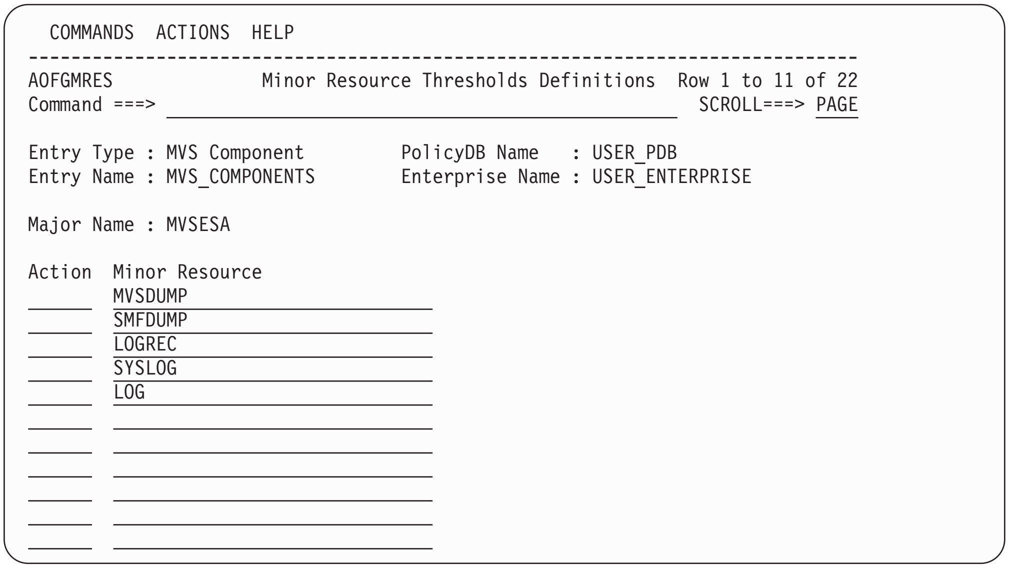
To add a minor resource enter a name in one of the empty entries at the bottom of the list.

To rename a minor resource overtype the current name with a new name and press Enter.

SA z/OS provides automation routines for recovery actions that use the automation flags that are defined for these minor resources together with commands that are defined for messages with the same name. For more information see the chapter “Automation Routines” in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

MINOR RESOURCE THRES Policy Item

The Minor Resource Thresholds Definitions panel, as shown in Figure 215, is displayed if you select the MINOR RESOURCE THRES policy item in the Policy Selection panel for the MVS Component entry type.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGMRES              Minor Resource Thresholds Definitions  Row 1 to 11 of 22
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : MVS Component         PolicyDB Name   : USER_PDB
Entry Name : MVS_COMPONENTS        Enterprise Name : USER_ENTERPRISE

Major Name : MVSESA

Action  Minor Resource
______  MVSDUMP
______  SMFDUMP
______  LOGREC
______  SYSLOG
______  LOG
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
______  ______________________________
```

Figure 215. Minor Resource Thresholds Definitions Panel

This panel allows you to manage thresholds for minor resources.

Major Name
This field shows the major resource name.

Action
Use this field to select (S) a minor resource for further processing or to delete (D) a minor resource.

Minor Resource
Use this field to specify the minor resource name according to the following rules:

- The name cannot contain any of the following characters: comma, single quotation mark, double quotation mark, left or right parenthesis, equal sign or a blank.
- The name cannot start or end with a period.
- The name cannot contain consecutive periods.

To add a minor resource enter a name in one of the empty entries at the bottom of the list.

To rename a minor resource overtype the current name with a new name and press Enter.

See also “Thresholds for Minor Resources” on page 20.

You can use special minor resource names for MVS Component processing, such as:

- MVSDUMP for system messages that report that an SVC dump has been made (IEA611E and IEA911E)
- SMFDUMP for system messages that report that the smfdump data set is full or has been switched (IEE362A, IEE362I, IEE391A, and IEE392I)
- LOGREC for system messages that report that the logrec data set is full or nearly full (IFB040I, IFB060E, IFB080E, IFB081I, and IFC001I)
- SYSLOG for system messages that report that the syslog has been queued (IEE043I)
- LOG for system messages that report that the syslog has become inactive (IEE037D—followed by IEE043I, IEE533E or IEE769E—and IEE041I)

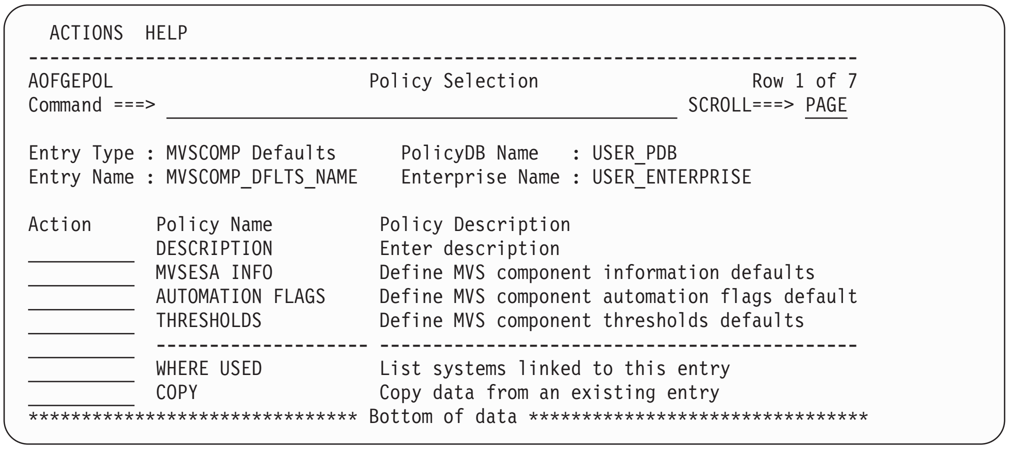
SA z/OS provides automation routines for recovery actions that use the thresholds that are defined for these minor resources together with commands that are defined for messages with the same name. For more information see the chapter “Automation Routines” in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

MVSCOMP Defaults Policy Object

MVSCOMP Defaults allow you to set default values for MVS Component sets.

Policy Items for MVSCOMP Defaults

Figure 216 shows the policy items that are displayed when a new MVSCOMP Defaults entry is defined or when existing MVSCOMP Defaults entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                    Row 1 of 7
Command ===> ______________________________________________ SCROLL===> PAGE

Entry Type : MVSCOMP Defaults      PolicyDB Name   : USER_PDB
Entry Name : MVSCOMP_DFLTS_NAME    Enterprise Name : USER_ENTERPRISE

Action      Policy Name         Policy Description
            DESCRIPTION         Enter description
__________  MVSESA INFO         Define MVS component information defaults
__________  AUTOMATION FLAGS    Define MVS component automation flags default
__________  THRESHOLDS          Define MVS component thresholds defaults
__________  -------------------- ----------------------------------------
__________  WHERE USED          List systems linked to this entry
__________  COPY                Copy data from an existing entry
__________
****************************** Bottom of data ********************************
```

Figure 216. Policy Selection Panel for MVSCOMP Defaults

The unique policy items for MVSCOMP Defaults are:

MVSESA INFO
This option allows you to specify a captured messages limit as the maximum number of messages that will be captured for each MVSESA resource while SA z/OS is running. These messages will show up in the

SA z/OS command DISPSYS. If the field is left blank the value can be inherited from the system defaults definition. If neither of them is specified, a default value of 0 is used.

Inform List can be used to specify where each MVSESA resource is registered to and who all status changes are propagated to. Valid receivers are SDF, IOM, and NMC. You can specify one or more values. If the field is left blank the value can be inherited from the systems default definition. If the resource should not be registered at all, specify NONE. This will prevent inheritance from systems default definitions.

AUTOMATION FLAGS
This option allows you to specify automation settings that will be used as defaults for MVS Components. The settings specified here will be used if you do not provide specific automation settings for MVS Components. See "Automation Flags" on page 20 for more information about setting automation flags.

THRESHOLDS
This policy item is used to set default error thresholds for MVS Components. The settings specified here will be used if you do not provide specific thresholds for MVS components. See "How SA z/OS Uses Error Thresholds" on page 17 for more information about error thresholds.

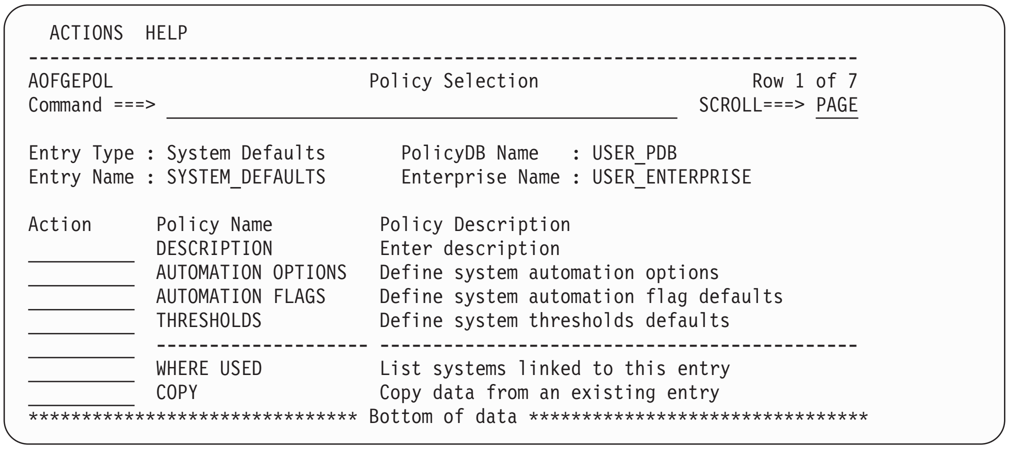
More than one MVSCOMP Default may be attached to a system, but it is recommended that you attach only one to avoid data being overwritten during SA z/OS initialization or an INGAMS REFRESH.

System Defaults Policy Object

This policy object allows you to set various defaults that are used throughout the system. The values set here may be overridden by other objects, such as Application class, Application Defaults, or MVSCOMP Defaults, or by setting appropriate values in individual application or MVS Component objects.

Policy Items for System Defaults

Figure 217 shows the policy items that are displayed when a new System Defaults entry is defined or when an existing System Defaults entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                    Row 1 of 7
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : System Defaults       PolicyDB Name   : USER_PDB
Entry Name : SYSTEM_DEFAULTS       Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  AUTOMATION OPTIONS   Define system automation options
__________  AUTOMATION FLAGS     Define system automation flag defaults
__________  THRESHOLDS           Define system thresholds defaults
__________  -------------------- ----------------------------------------------
__________  WHERE USED           List systems linked to this entry
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

Figure 217. Policy Selection Panel for System Defaults

The unique policy items for System Defaults policy object are:

AUTOMATION OPTIONS
: This policy item is used to set system-wide options. See “AUTOMATION OPTIONS Policy Item” for more information.

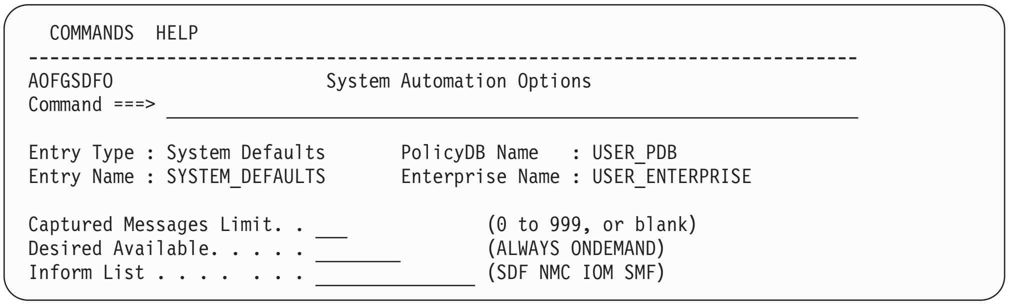
AUTOMATION FLAGS
: This policy item allows you to specify automation settings that are used as defaults for the system. The settings specified here are used if you do not provide specific automation settings for those policy objects that allow automation. See “Automation Flags” on page 20 for more information on setting automation flags.

THRESHOLDS
: This policy item is used to set system-wide error thresholds. The settings specified here are used if you do not provide object-specific threshold settings. See “How SA z/OS Uses Error Thresholds” on page 17 for more information on error thresholds.

More than one set of System Defaults may be attached to a system, but it is recommended that you attach only one to avoid data being overwritten during SA z/OS initialization or an INGAMS REFRESH.

AUTOMATION OPTIONS Policy Item

The System Automation Options panel, as shown in Figure 218, is displayed if you select the AUTOMATION OPTIONS policy item on the Policy Selection panel for the System Defaults entry type.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGSDFO                    System Automation Options
Command ===> _________________________________________________________________

Entry Type : System Defaults       PolicyDB Name   : USER_PDB
Entry Name : SYSTEM_DEFAULTS       Enterprise Name : USER_ENTERPRISE

Captured Messages Limit. . ___             (0 to 999, or blank)
Desired Available. . . . . ________        (ALWAYS ONDEMAND)
Inform List . . . . .  . . _______________ (SDF NMC IOM SMF)
```

Figure 218. System Automation Options Panel

The following parameters can be specified for system-wide defaults. If the corresponding entry is not specified for individual resources, this value will prevail.

Captured Messages Limit
: This specifies the maximum number of messages that will be captured for each resource while SA z/OS is running. This messages will show up in the DISPINFO or DISPMTR command output or SA z/OS Graphic Interface.

Desired Available
: Specifies how the desired state of the resource is determined in the absence of any propagated DesiredStatus votes on the resource.

You can specify the following values:

ALWAYS
: The resource's desired status is set to AVAILABLE. The resource will be started once its dependencies are satisfied. This is the same behavior that older releases of SA z/OS exhibit.

ONDEMAND

The resource's desired status is set to UNAVAILABLE. The resource will not be started. For SA z/OS to start the resource it must be propagated a MakeAvailable vote.

Note: If nothing is specified the default value is ALWAYS.

Inform List

This field allows you to specify:

- Where the application is registered to
- Where all status changes are propagated to
- Which SMF records are written

Valid receivers are SDF, NMC, IOM, and SMF. You can specify more than one receiver. The list of values can be blank or comma separated.

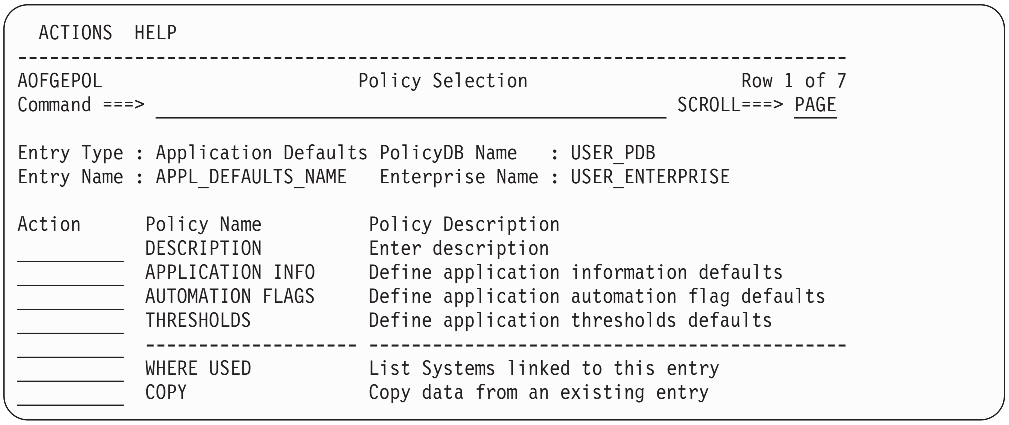
Neither system automation symbols (AOCCLONEs) nor system symbols are allowed.

Application Defaults Policy Object

The Application Defaults policy object specifies default values for automation flags and error threshold settings. These values will be used as the default for any Application object on the system that the Application Defaults set is attached to. If the application belongs to an application class with automation flags or error thresholds specified, the class policy takes precedence over the defaults.

Policy Items for Application Defaults

Figure 219 shows the policy items that are displayed when a new Application Defaults entry is defined or when an existing Application Defaults entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection                   Row 1 of 7
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Application Defaults PolicyDB Name   : USER_PDB
Entry Name : APPL_DEFAULTS_NAME   Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            DESCRIPTION          Enter description
__________  APPLICATION INFO     Define application information defaults
__________  AUTOMATION FLAGS     Define application automation flag defaults
__________  THRESHOLDS           Define application thresholds defaults
__________  -------------------- --------------------------------------------
__________  WHERE USED           List Systems linked to this entry
__________  COPY                 Copy data from an existing entry
```

Figure 219. Policy Selection Panel for Application Defaults

The unique policy items for Application Defaults policy are:

APPLICATION INFO

This policy item allows you to set various defaults for applications that you define to the system. The values entered in the Application Information Defaults panel are used as default values for Application objects that are attached to this system and do not have this information specified. For more details about these parameters, refer to “Application Policy Object” on page 156.

AUTOMATION FLAGS

This policy item allows you to specify automation flag settings that are used as defaults for all applications on the system. The settings specified here are used if you do not provide application-specific automation settings. See “Automation Flags” on page 20 for more information on setting automation flags.

THRESHOLDS

This policy item allows you to specify threshold settings that are used as defaults for all applications on the system. The settings specified here override system default settings and are used if you do not provide application-specific threshold settings. See “How SA z/OS Uses Error Thresholds” on page 17 for more information on setting thresholds.

More than one set of Application Defaults may be attached to a system, but it is recommended that you attach only one to avoid data being overwritten during SA z/OS initialization or an INGAMS REFRESH.

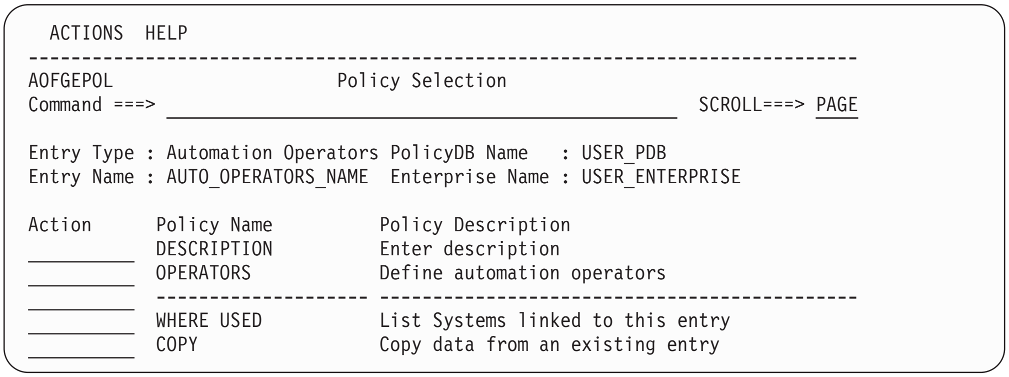
Automation Operators Policy Object

Automation operators are automated operator tasks, that can respond to messages from the operating system, applications, and network without requiring a human operator. Sets of automation operators can be linked to a system. Each automation operator task takes actions similar to those of a human operator, such as monitoring the system and handling and coordinating recovery and shutdown operations.

Automation operators are assigned specific messages to act on. During automated operations, the messages are routed to the appropriate operator using the NetView ASSIGN command.

Policy Items for Automation Operators

Figure 220 shows the policy items that are displayed when a new Automation Operators policy object is created or Automation Operators is selected from the Entry Type Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                    Policy Selection
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Automation Operators PolicyDB Name   : USER_PDB
Entry Name : AUTO_OPERATORS_NAME  Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  OPERATORS            Define automation operators
__________  -------------------- ---------------------------------------------
__________  WHERE USED           List Systems linked to this entry
__________  COPY                 Copy data from an existing entry
```

Figure 220. Policy Selection Panel for Automation Operators

The unique policy item for Automation Operators is:

OPERATORS

Automation operators are crucial to the operation of SA z/OS. The SA z/OS sample automation control file supplies definitions that are required in order to start SA z/OS. If you modify the sample definitions, or you do not want to use them, you need to be aware of which automation operators are required, and what messages should be assigned to them.

SA z/OS defines an *automated function name* for each automation operator that it uses. Certain automated function names must be defined. These are listed in Table 7. The automated function names are included in the sample policy databases that are supplied with SA z/OS. If you are not using a sample policy database, you must include these automated function names in your policy database. The automation operators that they are associated with are defined in the sample NetView DSIOPF data set.

OPERATORS Policy Item

Table 7 lists the automated function names that need to be defined.

Table 7. SA z/OS Mapping of NetView Automation Operator to SA z/OS Automated Function

Automated Function	Automation Operator	Description
AUTO1	AUTO1	Used for SA z/OS initialization
AUTO2	AUTO2	Used for SA z/OS initialization
BASEOPER	AUTBASE	Base automation: Backs up all other automation operators.
SYSOPER	AUTSYS	System automation: Acts as the primary system automation operator.
LOGOPER	AUTLOG	Used for message writing and for unassigned messages.
MSGOPER	AUTMSG	Message automation: Formats and routes messages.
AOFWRK*xx*	AUTWRK*xx*	Used for subsystem processing, for example startup, shutdown
NETOPER	AUTNET1, AUTNET2	VTAM and NetView automation: Acts as the network products automation operator.
JESOPER	AUTJES	JES automation: Acts as the JES2 or JES3 product automation operator.
MONOPER	AUTMON	Monitoring and startup automation: Performs active monitoring of resources.
RECOPER	AUTREC	Recovery automation: Acts as the primary recovery operator. It is not recommended that you use RECOPER for user automation that issues WTO/Rs. This situation can halt SA z/OS's WTO/R buffer recovery process, which may result in a buffer full situation.
RPCOPER	AUTRPC	Used for XCF requests
SHUTOPER	AUTSHUT	Shutdown automation: Acts as the base shutdown operator.

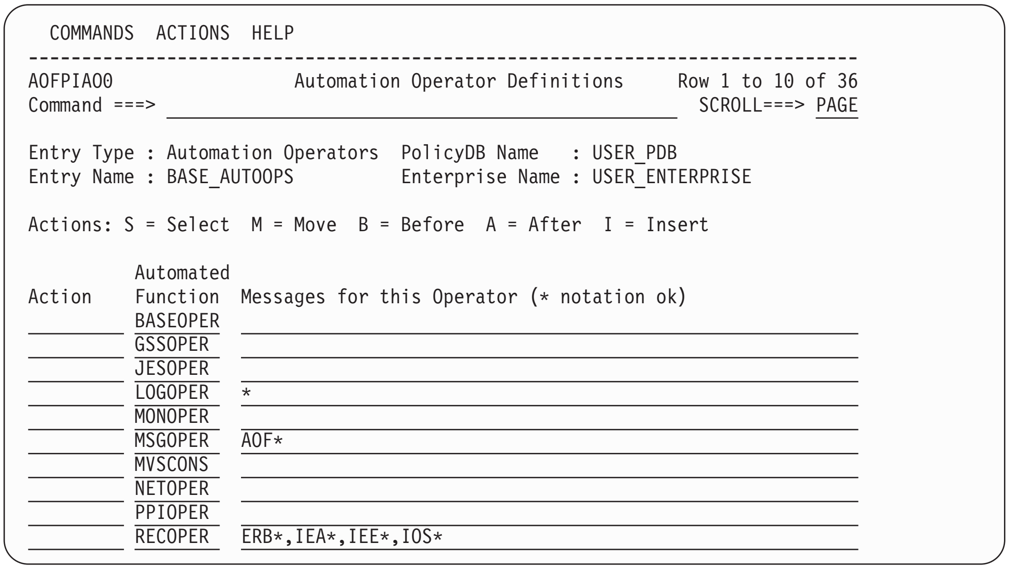
Table 7. SA z/OS Mapping of NetView Automation Operator to SA z/OS Automated Function (continued)

Automated Function	Automation Operator	Description
GSSOPER	AUTGSS	General automation. Acts as the operator for applications using SA z/OS generic automation routines, but does not execute these routines. Instead, it distributes them to the responsible AOFWRKxx automated functions. GSSOPER will also be used as backup in case there is no AOFWRKxx automated function defined.
GATOPER	GAT&DOMID	Gateway: Acts as the outbound gateway operator for automation notification message forwarding
MVSCONS*i*	AUTCON	Used for the MVS and NetView console interface.
PPIOPER	AOFARCAT	ARM PPI receiver
HBOPER	AUTHB	Used by heartbeat function
XCFOPER	AUTXCF	Used for enhanced Parallel Sysplex automation functions.
XCFOPER2	AUTXCF2	Used for enhanced Parallel Sysplex automation functions.
HWOPER1–33	AUTHW1–33	Used for Hardware Validation messages.
PLEXOPER	AUTPLEX	Used for enhanced Parallel Sysplex automation functions.
PLEXOPR2	AUTPLEX2	Used for enhanced Parallel Sysplex automation functions.
PLEXOPR3	AUTPLEX3	Used for enhanced Parallel Sysplex automation functions.
POSTOPER	AUTPOST	Used for the serialization of INGPOST commands on the target system.
POSTSLV	AUTPOSTS	Used for the serialization of INGPOST commands on the focal point.
AOFSES*xx*	AUTSES*xx*	Used for OMEGAMON® sessions.
EVTOPER	AUTEVT1, AUTEVT2	Used for event subscription and the TWS command request interface.

It is highly recommended that you assign the suggested automated operators to these required automated functions as listed in Table 7 on page 292. If you attempt to share automated operators across more than one automated function you will experience a significant performance degradation during SA z/OS initialization or an INGAMS REFRESH.

Sample DSIOPF definition statements are included in the AOFOPF* samples that are shipped with SA z/OS. You may define additional automation operators, to suit your automation needs.

When defining SA z/OS automated functions, you must associate the SA z/OS automated function names with actual automation operator IDs defined in member DSIOPF in the NetView DSIPARM data set. For example, you can associate the SA z/OS name BASEOPER with the NetView automation operator ID AUTBASE, MSGOPER with AUTMSG, and JESOPER with AUTJES. You must also define synonyms for them in AOFMSGSY. See *IBM Tivoli System Automation for z/OS Customizing and Programming* for more information on synonyms. Figure 221 shows an example of an Automation Operator Definition panel.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFPIAO0                 Automation Operator Definitions     Row 1 to 10 of 36
Command ===>                                                  SCROLL===> PAGE

Entry Type : Automation Operators  PolicyDB Name   : USER_PDB
Entry Name : BASE_AUTOOPS          Enterprise Name : USER_ENTERPRISE

Actions: S = Select  M = Move  B = Before  A = After  I = Insert

          Automated
Action    Function  Messages for this Operator (* notation ok)
          BASEOPER
          GSSOPER
          JESOPER
          LOGOPER   *
          MONOPER
          MSGOPER   AOF*
          MVSCONS
          NETOPER
          PPIOPER
          RECOPER   ERB*,IEA*,IEE*,IOS*
```

Figure 221. Automation Operator Definition Panel

Messages are entered in the **Messages for this Operator** field. You can specify:

- A single message, such as IEE115I
- A group of messages, by specifying a message prefix and an asterisk (*), such as AOF* or $HASP*
- All messages, not otherwise assigned, by specifying an asterisk (*) only

The order of the automation operators in the list is irrelevant for the execution. NetView ensures that the messages are assigned correctly with the most specific first, followed by the more general. You can use the following action codes to change the order of automation operators in the list:

(M) MOVE
Moves the automation operator to another position in the list that is marked with BEFORE and AFTER.

(B) BEFORE
Is used with MOVE. The automation operator is placed before the line marked with the BEFORE action.

(A) AFTER
Is used with MOVE. The automation operator is placed after the line marked with the AFTER action.

(I) INSERT
Inserts a new line directly underneath the current one. You can add another automation operator or message on this line.

To continue the specification of message IDs across several lines, insert a line where required and leave the automated function blank.

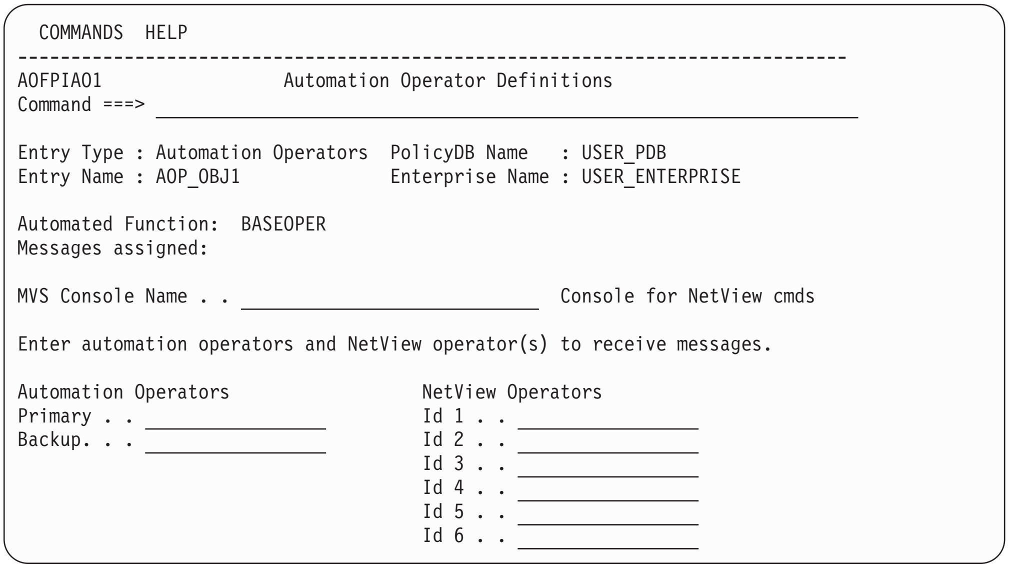
The following automated functions should not have messages assigned to them on the Automation Operator Definition panel:

- BASEOPER

- GATOPER
- MVSCONS*i* (where *i* is blank, A-Z, or 0-9)
- SHUTOPER
- HBOPER
- AOFWRKxx
- GSSOPER

For these automation operators, you should only specify the automated function name in the Automated Function field, and leave the Messages for this Operator field blank.

After you have entered the automated function and assigned messages, a new panel is displayed where you can enter the automation operators and NetView operators that are to receive messages. Figure 222 provides an example.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIAO1                 Automation Operator Definitions
Command ===> __________________________________________________________________

Entry Type : Automation Operators  PolicyDB Name   : USER_PDB
Entry Name : AOP_OBJ1              Enterprise Name : USER_ENTERPRISE

Automated Function:  BASEOPER
Messages assigned:

MVS Console Name . . ____________________________  Console for NetView cmds

Enter automation operators and NetView operator(s) to receive messages.

Automation Operators                  NetView Operators
Primary . . _________________         Id 1 . . _________________
Backup. . . _________________         Id 2 . . _________________
                                      Id 3 . . _________________
                                      Id 4 . . _________________
                                      Id 5 . . _________________
                                      Id 6 . . _________________
```

Figure 222. Automation Operator Definitions Panel

The fields are:

Automation Operators

Each automation operator relates to a specific NetView operator ID. Each message is assigned to a list of operators. This list includes the primary NetView ID, the secondary NetView ID (if specified), the primary NetView ID for SYSOPER, and the primary NetView ID for BASEOPER. The message is processed on the first operator in the list that is logged on when the message is received. If none of the operators are logged on, it will be processed on the task that started the system NetView Subsystem Interface Router task (SIR task), if it was started by a NetView task. In the NetView default initialization sequence, AUTO1 starts the SIR task. If the SIR task was not started by an autotask (that is, it has INIT=Y coded in DSIDMN) or the task that started it is no longer active, the message will not be automated. Similar lists of autotasks are defined in AOFMSGSY and are used to route commands to tasks from the NetView automation table.

If a group is specified (the ID starts with a “+”), all the IDs that are assigned to the group are inserted into the list at that point.

You may also use SYSCLONE variables and the &DOMAIN. variable for the automation operators and NetView operators specification.

MVS Console Name

This is a 1-8 alphanumeric field as defined in the CONSOL*nn* member of the MVS PARMLIB. The system console name *MASTER* or *ANY* can also be used. Otherwise, the first character of the name must be alphabetic. If console name is specified, the autotask is considered as an attended task. It will be applied to the Primary Automation Operator only.

The name may include any system symbol (for example, `&DOMAIN.`). System automation symbols (`&AOCCLONEx.`) are not allowed.

NetView Operators

Defines NetView IDs to receive copies of the messages assigned to the automation operator. The ID must be logged on to receive the message and either the primary or backup operator must also be logged on. Operator ″groups″ can be defined by specifying a ″+″ as the first character.

Structure of AOFWRKxx Operators

AOFWRKxx operators are used to distribute the work on different NetView tasks, so that they can run in parallel. AOFWRKxx operators are used as dedicated automated function responsible for all the work on a particular subsystem. However, since there might be too many subsystems defined, it may not be possible that each subsystem has a unique automated function for exclusive usage. Instead, several subsystems will share the same automated function. This automated function is responsible for the subsystem's startup, shutdown, monitoring and message handling activities.

You must define at least three automated functions which are responsible for the subsystem automation. You have the possibility to define additional ones. So you can decide how much parallelism you want.

The naming of these automation operator entries is of type `AOFWRKxx` where `xx` is any number or character. The `xx` part of the automated function name does not have to be in sequence.

At SA z/OS initialization or an INGAMS REFRESH, each defined subsystem will be assigned an automated function from the pool of defined `AOFWRKxx` operators.

Behavior of SHUTDOWN, STARTUP and GSSOPER

There are changes in the behavior of the shutdown, the startup and in the usage of the GSSOPER automated function.

SHUTDOWN and STARTUP: A shutdown/startup can continue to process another subsystem on another automated function before a shutdown/startup command is completed. The shutdown/startup process can run in parallel.

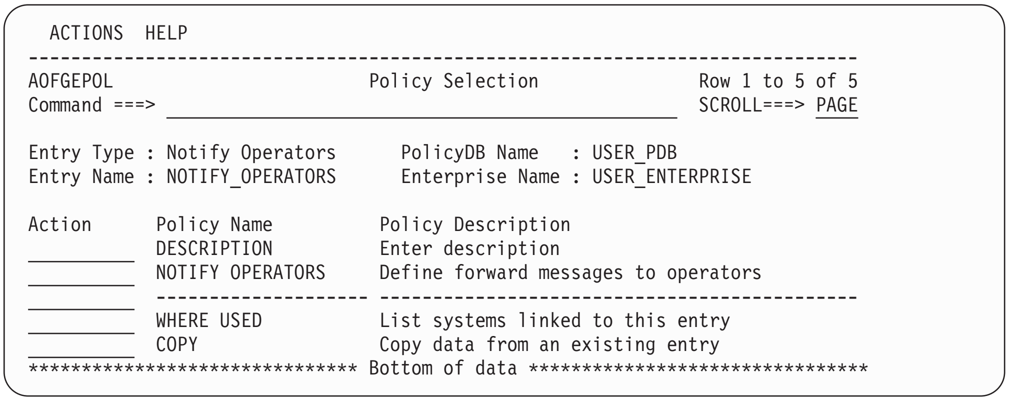
GSSOPER Usage: The GSSOPER automated function is used for the generic routines ACTIVMSG, TERMMSG and HALTMSG. These routines are routed from the message automation table to GSSOPER. The GSSOPER automated function will no longer execute these routines, it will only distribute them to the responsible AOFWRKxx automation operators. GSSOPER will also be used as backup when there is no AOFWRKxx automation operator defined.

Notify Operators Policy Object

The Notify Operators policy object allows you to define notification messages for certain operators.

Policy Items for Notify Operators

Figure 223 shows the Entry Selection panel to define these operators.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection             Row 1 to 5 of 5
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Notify Operators      PolicyDB Name   : USER_PDB
Entry Name : NOTIFY_OPERATORS      Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  NOTIFY OPERATORS     Define forward messages to operators
__________  -------------------- ---------------------------------------------
__________  WHERE USED           List systems linked to this entry
__________  COPY                 Copy data from an existing entry
****************************** Bottom of data ********************************
```

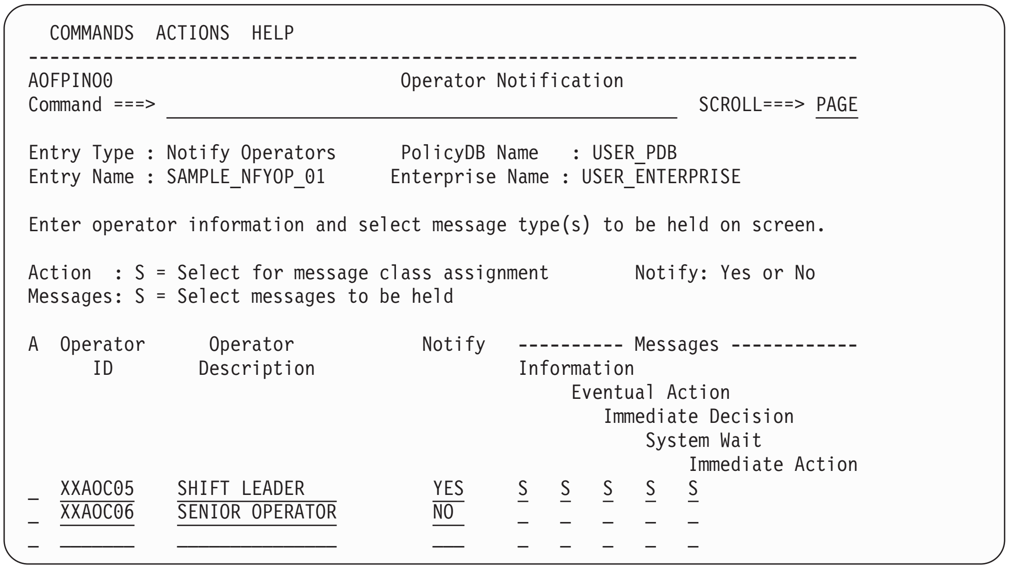
Figure 223. Policy Selection Panel for Notify Operators

NOTIFY OPERATORS

Notification operators are human NetView operators who receive SA z/OS notification messages, which are messages about events SA z/OS detects or actions it takes. These notification messages are routed to one or more logged-on human operators, using the message forwarding path you set up in the Network policy object. The messages can optionally be held until the operator clears them.

NOTIFY OPERATORS Policy Item

Figure 224 shows the Operator Notification panel that you can use to select notification operators or define new notification operators that you want to be notified about selected message classes from applications on a system.

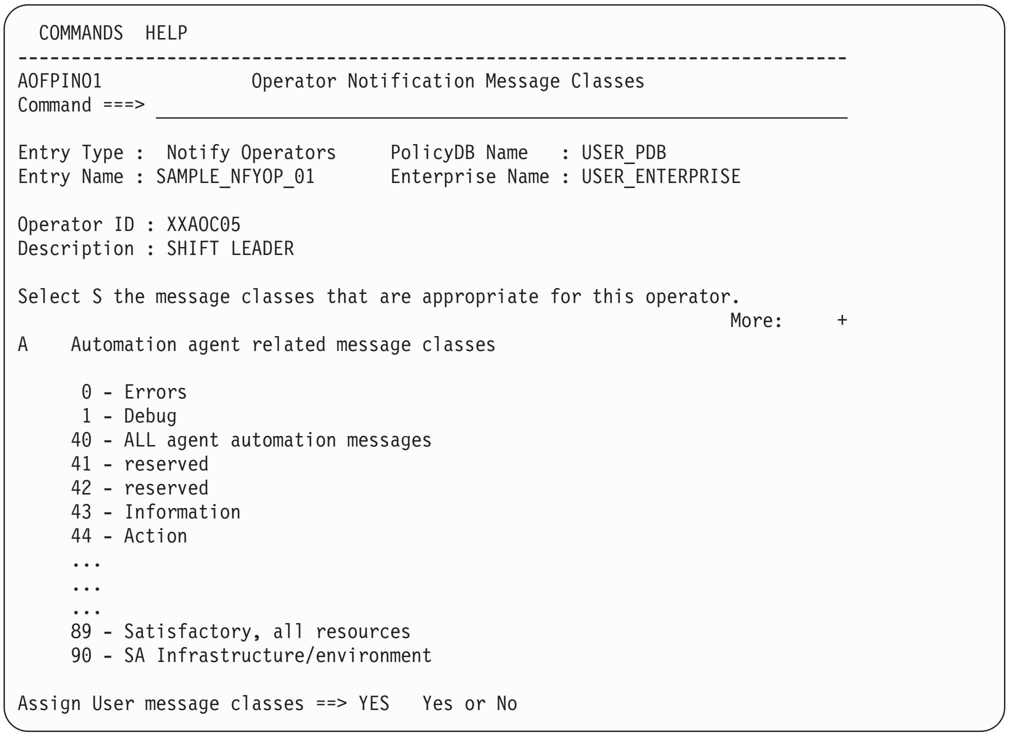


```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFPINO0                           Operator Notification
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Notify Operators      PolicyDB Name   : USER_PDB
Entry Name : SAMPLE_NFYOP_01      Enterprise Name : USER_ENTERPRISE

Enter operator information and select message type(s) to be held on screen.

Action  : S = Select for message class assignment       Notify: Yes or No
Messages: S = Select messages to be held

A  Operator      Operator           Notify   ---------- Messages ------------
      ID       Description                   Information
                                                  Eventual Action
                                                     Immediate Decision
                                                         System Wait
                                                             Immediate Action
_  XXAOC05    SHIFT LEADER          YES     S   S   S   S   S
_  XXAOC06    SENIOR OPERATOR       NO      _   _   _   _   _
_  _______    _______________       ___     _   _   _   _   _
```

Figure 224. Operator Notification Panel

When you specify *Yes* for `Assign User message classes`, to have the operator notified about user-defined message classes, SA z/OS displays existing message classes that have been defined with policy item *Auto Msg Classes* of entry type ENT.

How to define messages for a message class and where to store them, is described in *IBM Tivoli System Automation for z/OS Customizing and Programming*.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPINO1              Operator Notification Message Classes
Command ===> _________________________________________________________________

Entry Type :  Notify Operators     PolicyDB Name   : USER_PDB
Entry Name : SAMPLE_NFYOP_01       Enterprise Name : USER_ENTERPRISE

Operator ID : XXAOC05
Description : SHIFT LEADER

Select S the message classes that are appropriate for this operator.
                                                                   More:     +
A    Automation agent related message classes

      0 - Errors
      1 - Debug
     40 - ALL agent automation messages
     41 - reserved
     42 - reserved
     43 - Information
     44 - Action
     ...
     ...
     ...
     89 - Satisfactory, all resources
     90 - SA Infrastructure/environment

Assign User message classes ==> YES   Yes or No
```

Figure 225. Operator Notification Message Classes Panel

When defining notification operators, you can customize:

- Which operators receive notification messages
- The types of message held on notification operators' console screens:
  - Immediate Action
  - Immediate Decision
  - Possible Action
  - Information
  - System Wait
- The classes of messages sent to the operators. (See "AUTO MSG CLASSES Policy Item" on page 96 for information about defining your own message classes.) Each SA z/OS message can have one or more message classes associated with it. Similarly, each notification operator can have one or more message classes assigned. When messages are processed during automated or manual operations, the class of each message is compared to the classes defined for each notification operator. If a match occurs, the notification operator receives the message. All one and two digit numeric message classes are reserved for IBM use. You may use message classes made up of one alphabetic and one numeric character (or one numeric and one alphabetic).

  Table 8 on page 299 shows the message classes defined by SA z/OS. Message classes from 80 to 90 are relevant for monitoring the automation manager. These message classes are discussed in more detail in *IBM Tivoli System Automation for z/OS Messages and Codes*.

Table 8. SA z/OS Message Classes

Message Class Description	**Message Class Number**
General Message Classes	
Errors	0
Debug	1
Automation Agent-Related Message Classes	
All automation agent messages	40
Information	43
Action	44
Status	45
User intervention	46
Critical threshold setting exceeded	50
Frequent threshold setting exceeded	51
Infrequent threshold exceeded	52
IMS Automation Messages	60
Automation Manager-Related Message Classes	
All automation manager messages	80
Intervention required	81
Automation denied	82
Automation Impacted	83
Awaiting Automation	84
In Automation	85
Degraded - Target resources	86
Degraded - All resources	87
Satisfactory - Target resources	88
Satisfactory - All resources	89
SA z/OS Infrastructure/Environment	90

For example, any operator defined with a notification operator class of all SA z/OS messages (class number 40 and 80) receives all SA z/OS messages. Operators responsible for system operations should receive all SA z/OS messages or a subset of those messages. As another example, you may want to have your lead operator receive all debug messages.

SA z/OS uses the general and console automation message classes only. The SA z/OS customization dialog supports the message classes used by SA z/OS only.

SA z/OS uses notification messages to update the status display facility (SDF).

Network Policy Object

When multiple z/OS systems are interconnected and require consolidated operations at one focal point system, you can use focal point services to monitor and control the systems. These services allow an operator to view the status of multiple systems from a single system acting as a focal point and to send messages, commands, and responses from a focal point system to other systems. The Network policy object allows you to:

- Define paths for sending messages, commands, or responses from one system to another.
- Define primary and backup focal point systems.
- Define user Terminal Access Facility (TAF) fullscreen sessions and specify whether each of these should be initiated automatically when the operator logs on to the SA z/OS NetView Interface. If a TAF session is not defined to start automatically, you can initialize it using the OPER dialog.
- Define gateway sessions connecting systems.

  A gateway session uses autotasks both inbound to and outbound from a system to send data between systems.

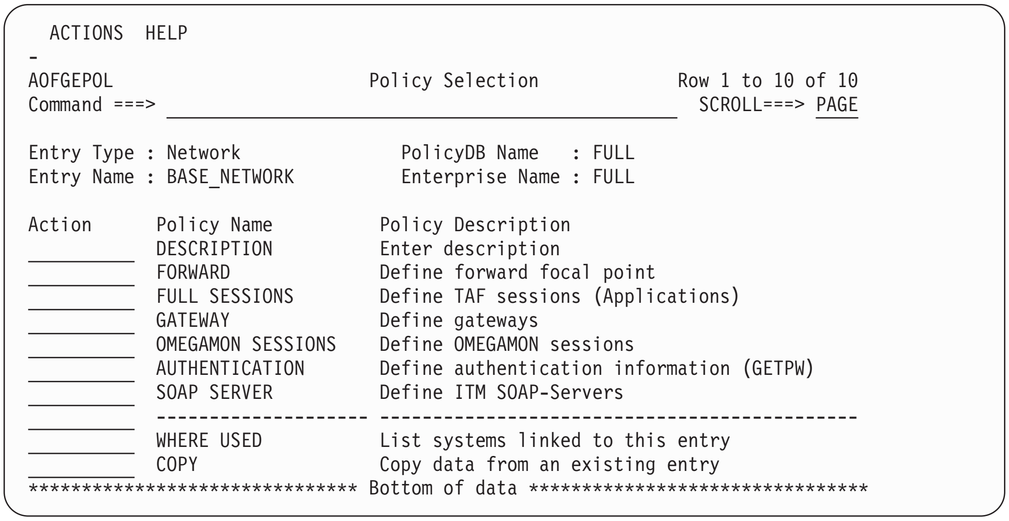
  Communication with the focal point system is performed using the NetView RMTCMD command or XCF. XCF is used when commands are sent to systems within the same sysplex (that is, with the same XCF group ID). Otherwise, RMTCMD is used. In both cases, you can see the responses on your operator task.
- Define OMEGAMON sessions.

  OMEGAMON sessions can be used to retrieve exception information or general performance and availability data for health monitoring and proactive automation.
- Manage access keys to the NetView password data set, where passwords can be stored that are needed when OMEGAMON sessions have to be established.
- Define Simple Object Access Protocol (SOAP) servers and their attributes to be able to communicate with IBM Tivoli Monitoring (ITM).

Chapter 10, “Automating Networks,” on page 375 provides more information on the defining automation networks.

Policy Items for Network

Figure 226 shows the policy items that are displayed when a new Network policy object is defined or when an existing Network object is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
-
AOFGEPOL                        Policy Selection             Row 1 to 10 of 10
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Network               PolicyDB Name   : FULL
Entry Name : BASE_NETWORK          Enterprise Name : FULL

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  FORWARD              Define forward focal point
__________  FULL SESSIONS        Define TAF sessions (Applications)
__________  GATEWAY              Define gateways
__________  OMEGAMON SESSIONS    Define OMEGAMON sessions
__________  AUTHENTICATION       Define authentication information (GETPW)
__________  SOAP SERVER          Define ITM SOAP-Servers
__________  -------------------- ---------------------------------------------
__________  WHERE USED           List systems linked to this entry
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

Figure 226. Policy Selection Panel for Network

The unique policy items for Network are:

FORWARD
This policy item is used to define the domain that is to receive the notification messages to be forwarded from the current domain. See "FORWARD Policy Item" for more details.

FULL SESSIONS
With this policy item you can define applications that TAF fullscreen sessions can be established to. See "FULL SESSIONS Policy Item" on page 302 for more details.

GATEWAY
Each domain that is to receive commands, replies, or notification messages must have a gateway defined for it. Communication with the focal point system is performed via XCF (within the same sysplex) or the NetView RMTCMD command.See "GATEWAY Policy Item" on page 302 for more details.

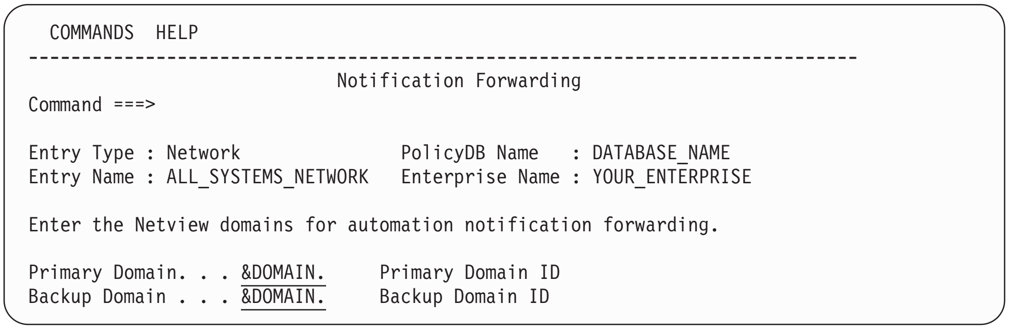
OMEGAMON SESSIONS
This policy item is used to define the attributes used to establish and run a communication session with OMEGAMON based on the NetView Terminal Access Facility (TAF). See "OMEGAMON Policy Item" on page 303 for more details.

AUTHENTICATION
This policy item is used to define access keys for particular OMEGAMON application IDs in order to let SA z/OS retain passwords in the NetView password data set. See "AUTHENTICATION Policy Item" on page 305 for more details.

SOAP SERVER
This policy item is used to define SOAP servers and their attributes. See "SOAP SERVER Policy Item" on page 306 for more details.

FORWARD Policy Item

If you select the FORWARD policy item from the Network Policy Selection panel, the panel shown in Figure 227 is displayed.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
                             Notification Forwarding
Command ===>

Entry Type : Network               PolicyDB Name   : DATABASE_NAME
Entry Name : ALL_SYSTEMS_NETWORK   Enterprise Name : YOUR_ENTERPRISE

Enter the Netview domains for automation notification forwarding.

Primary Domain. . . &DOMAIN.     Primary Domain ID
Backup Domain . . . &DOMAIN.     Backup Domain ID
```

Figure 227. Notification Forwarding Panel

This option is used to define the domain that is to receive the notification messages to be forwarded from the current domain. Two domains can be specified in the fields of the panel:

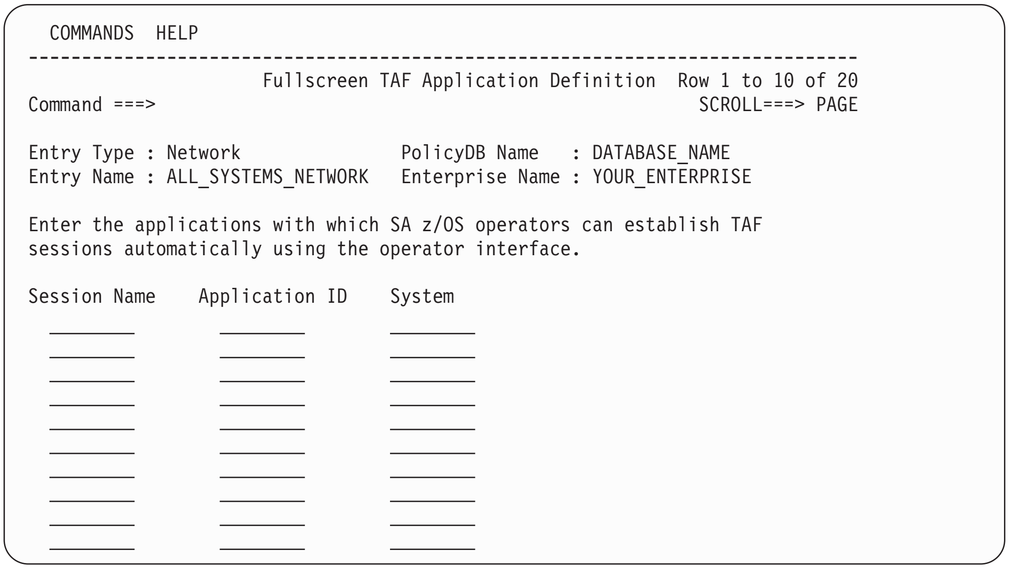
Primary Domain
In this field you enter the domain that is to receive the notification messages from the current domain. If this field is not filled in, the current domain will be the focal point (other domains forwarding messages to the

current one without it forwarding any itself). The name must consist of up to 5 alphanumeric characters or can be &DOMAIN..

Backup Domain
If the primary domain is not active, the domain defined here acts as the backup and receives its messages. If this field is not filled in, the backup defaults to the current domain. The name must consist of up to 5 alphanumeric characters or can be &DOMAIN..

FULL SESSIONS Policy Item

If you select the FORWARD policy item from the Network Policy Selection panel, the panel shown in Figure 228 is displayed.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
                      Fullscreen TAF Application Definition  Row 1 to 10 of 20
Command ===>                                                 SCROLL===> PAGE

Entry Type : Network               PolicyDB Name   : DATABASE_NAME
Entry Name : ALL_SYSTEMS_NETWORK   Enterprise Name : YOUR_ENTERPRISE

Enter the applications with which SA z/OS operators can establish TAF
sessions automatically using the operator interface.

Session Name    Application ID    System
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
```

Figure 228. Fullscreen TAF Application Definition Panel

With this option you can define applications that TAF fullscreen sessions can be established to. The following fields can be specified:

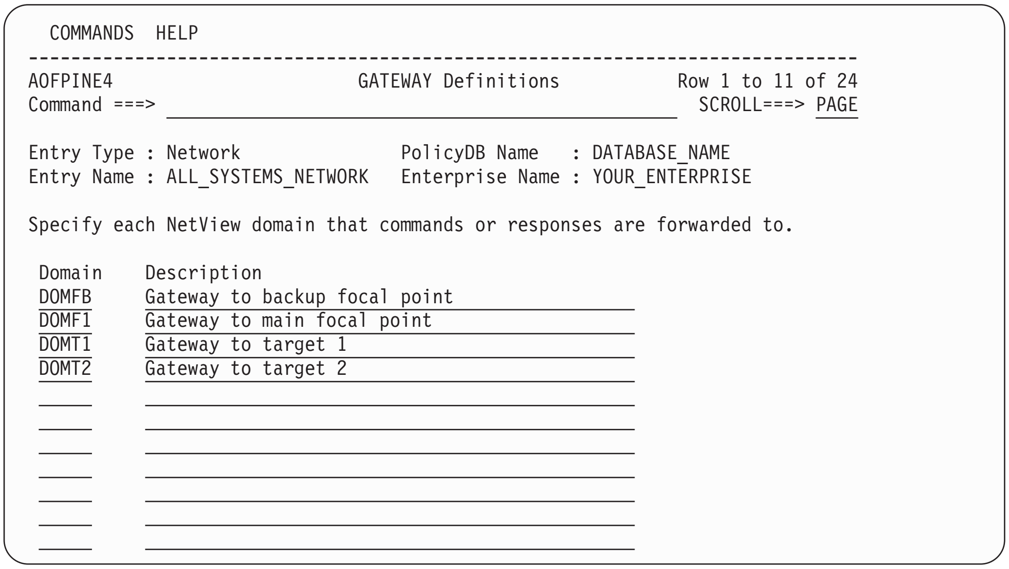
Session Name
This field acts as a description of the application that is displayed on the interface panel.

Application ID
The application name as specified in VTAMLST.

System
The name of the system that the application runs on. This field also acts as a description field that is displayed on the interface panel.

GATEWAY Policy Item

If you select the GATEWAY policy item from the Network Policy Selection panel, the panel shown in Figure 229 on page 303 is displayed.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPINE4                      GATEWAY Definitions          Row 1 to 11 of 24
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Network              PolicyDB Name   : DATABASE_NAME
Entry Name : ALL_SYSTEMS_NETWORK   Enterprise Name : YOUR_ENTERPRISE

Specify each NetView domain that commands or responses are forwarded to.

 Domain    Description
 DOMFB     Gateway to backup focal point
 DOMF1     Gateway to main focal point
 DOMT1     Gateway to target 1
 DOMT2     Gateway to target 2
 _____     ______________________________________________
 _____     ______________________________________________
 _____     ______________________________________________
 _____     ______________________________________________
 _____     ______________________________________________
 _____     ______________________________________________
 _____     ______________________________________________
```

Figure 229. GATEWAY Definitions Panel

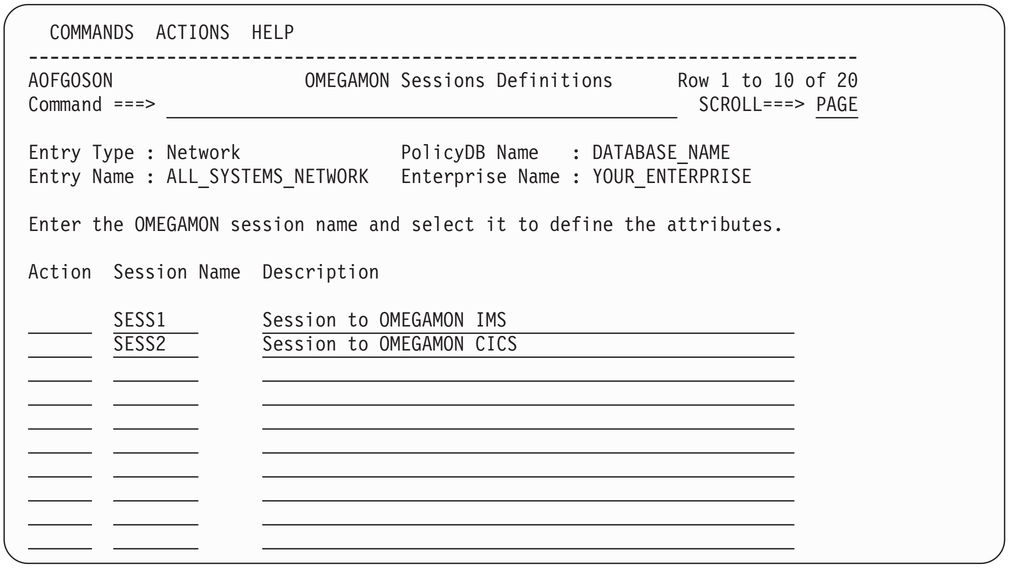
Each domain that is to receive commands, replies or notification messages must have a gateway defined for it. This includes the domain name for the outbound gateway autotask. To define the gateway, provide the following information:

Domain
: The name of all the domains that should receive anything forwarded from the current domain.

Description
: The description of the domain that is to be communicated with through the gateway.

OMEGAMON Policy Item

If you select the OMEGAMON policy item from the Network Policy Selection panel, the panel shown in Figure 230 is displayed.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGOSON                 OMEGAMON Sessions Definitions      Row 1 to 10 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Network              PolicyDB Name   : DATABASE_NAME
Entry Name : ALL_SYSTEMS_NETWORK   Enterprise Name : YOUR_ENTERPRISE

Enter the OMEGAMON session name and select it to define the attributes.

Action  Session Name  Description

______  SESS1         Session to OMEGAMON IMS
______  SESS2         Session to OMEGAMON CICS
______  ________      __________________________________________________
______  ________      __________________________________________________
______  ________      __________________________________________________
______  ________      __________________________________________________
______  ________      __________________________________________________
______  ________      __________________________________________________
______  ________      __________________________________________________
______  ________      __________________________________________________
```

Figure 230. OMEGAMON Sessions Definitions Panel

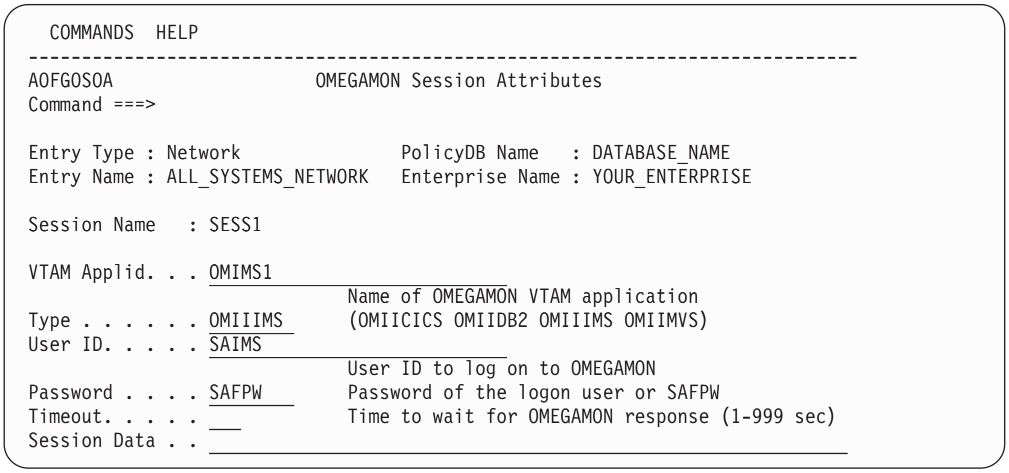
This panel allows you to create, select or delete OMEGAMON sessions.

Action
Can be used to select (S) a session for further processing or delete (D) a session.

Session Name
A unique, eight character name for the session that you are defining. Any alphabetic, numeric, #, @ and $ character can be used. The first character must be alphabetic.

Description
This is optional arbitrary text that describes the purpose of this session.

Selecting a session from the OMEGAMON Session Definitions panel leads to the OMEGAMON Session Attributes panel, as shown in Figure 231.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGOSOA                   OMEGAMON Session Attributes
Command ===>

Entry Type : Network              PolicyDB Name   : DATABASE_NAME
Entry Name : ALL_SYSTEMS_NETWORK   Enterprise Name : YOUR_ENTERPRISE

Session Name   : SESS1

VTAM Applid. . . OMIMS1
                             Name of OMEGAMON VTAM application
Type . . . . . . OMIIIMS     (OMIICICS OMIIDB2 OMIIIMS OMIIMVS)
User ID. . . . . SAIMS
                             User ID to log on to OMEGAMON
Password . . . . SAFPW       Password of the logon user or SAFPW
Timeout. . . . . ___         Time to wait for OMEGAMON response (1-999 sec)
Session Data . .
```

Figure 231. OMEGAMON Session Attributes Panel

This panel allows you to define OMEGAMON sessions together with their attributes.

VTAM Applid
This is the name of the OMEGAMON VTAM application as defined by the installation during customization of the OMEGAMON product. The name is 1 to 8 characters long and may contain system and automation symbols.

Type This is the type of OMEGAMON monitor that a session is established with. Valid types are OMIICICS, OMIIDB2, OMIIIMS, and OMIIMVS; they represent the OMEGAMON classic for CICS, DB2, IMS, and MVS, respectively.

User ID
This is the ID of the user that logs on to OMEGAMON when a session is established. To establish such a session, the user ID must be authorized to log on to OMEGAMON either within OMEGAMON itself or using an SAF-product.

To issue OMEGAMON commands, the user ID must also be authorized to issue commands that are protected by OMEGAMON internal command security or by the security profiles that are managed by an SAF product. For more details about user authorization, see the security chapter in *IBM Tivoli System Automation for z/OS Planning and Installation*.

The user ID has a maximum length of 8 characters and may contain system and automation symbols. It can be omitted if no OMEGAMON product level security is required.

Password

The password of the user who logs on to OMEGAMON. The password has a maximum length of 8 characters and must be specified if a user ID was specified.

The predefined keyword SAFPW may be used to allow SA z/OS to retain the password in the NetView password data set for the user ID. See also "AUTHENTICATION Policy Item."

Timeout

This is the maximum time in seconds that SA z/OS waits for a response from OMEGAMON. If the specified timeout expires, the user is logged off and the session is terminated.

Any value between 1 and 999 seconds can be specified. The default value is derived from the setting of the WAITTIME variable (default: 29 seconds) that can be set in CNMSTYLE or with the TIMEOUT VALUES policy item in the customization dialog (see "TIMEOUT VALUES Policy Item" on page 278).

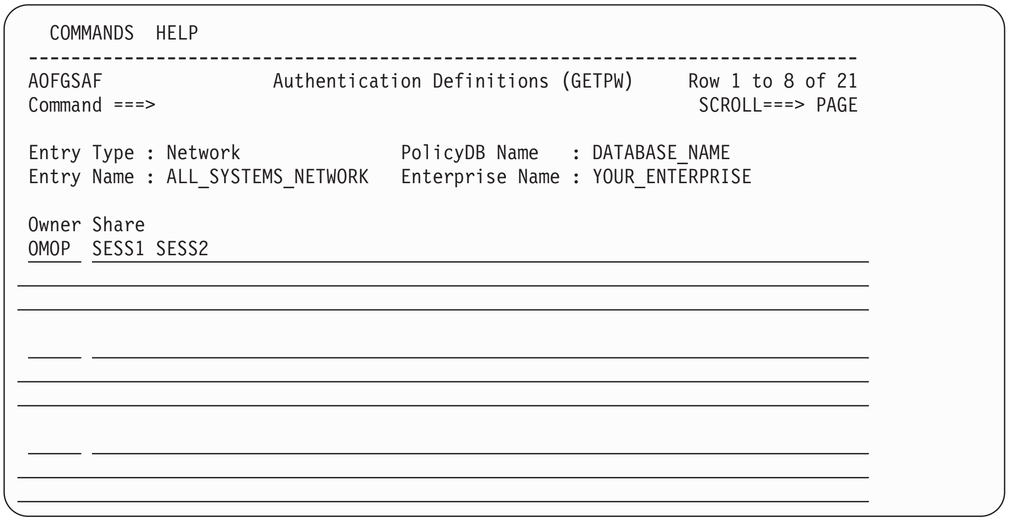
Session Data

This is an optional 60-character field of data that is passed unchanged to OMEGAMON during logon. The data field must be used to specify the job name of a particular CICS region or the name of a particular DB2 subsystem that should be monitored by automation, for example:

`CICS=PRDAOR1`	where `PRDAOR1` is the job name of the CICS region
`DB2=SGG4`	where `SGG4` denotes the DB2 subsystem

AUTHENTICATION Policy Item

If you select the AUTHENTICATION policy item from the Network Policy Selection panel, the panel shown in Figure 232 is displayed.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGSAF              Authentication Definitions (GETPW)     Row 1 to 8 of 21
Command ===>                                                 SCROLL===> PAGE

Entry Type : Network              PolicyDB Name   : DATABASE_NAME
Entry Name : ALL_SYSTEMS_NETWORK   Enterprise Name : YOUR_ENTERPRISE

Owner Share
OMOP  SESS1 SESS2
```

Figure 232. Authentication Definitions Panel

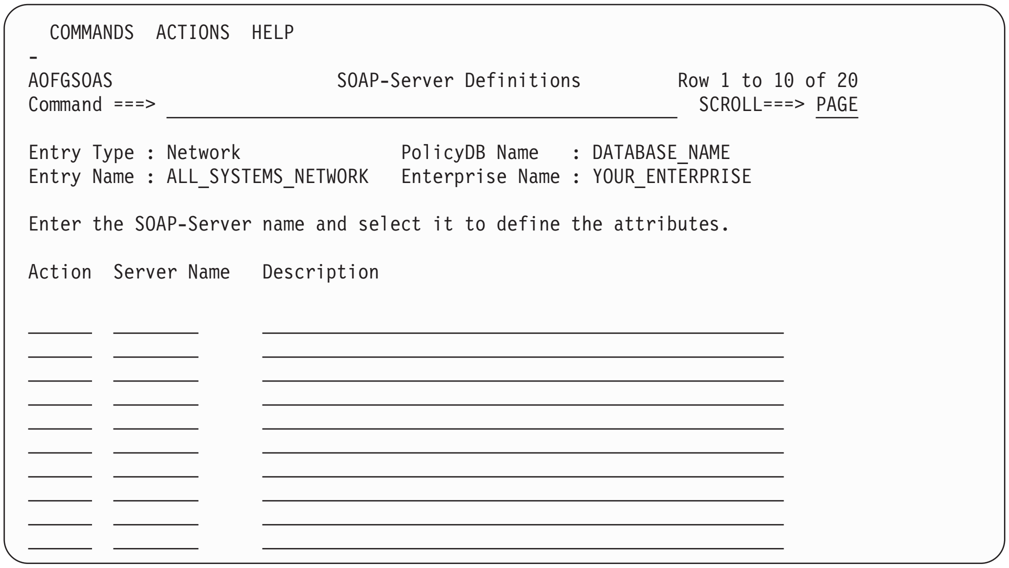
When SAFPW is specified as the password for a given OMEGAMON session, SA z/OS retrieves the actual password from the password data set using the user ID that is specified for this session and the owner that is specified in this policy item. For more details about the use of the NetView password data set, see the security chapter in *IBM Tivoli System Automation for z/OS Planning and Installation*.

Owner
Custom value of up to 5 characters that is used by SA z/OS to look up the password in the NetView password data set when an OMEGAMON session with any of the VTAM application IDs listed under **Share** is established.

Share
This is the list of VTAM application IDs representing OMEGAMON monitors that passwords are kept for in the NetView password data set under the same owner. The application IDs may be delimited by blank or comma.

SOAP SERVER Policy Item

If you select the SOAP SERVER policy item from the Network Policy Selection panel, the panel shown in Figure 233 is displayed.



```
  COMMANDS  ACTIONS  HELP
-
AOFGSOAS                    SOAP-Server Definitions        Row 1 to 10 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Network              PolicyDB Name   : DATABASE_NAME
Entry Name : ALL_SYSTEMS_NETWORK   Enterprise Name : YOUR_ENTERPRISE

Enter the SOAP-Server name and select it to define the attributes.

Action  Server Name   Description

______  ________      _________________________________________________
______  ________      _________________________________________________
______  ________      _________________________________________________
______  ________      _________________________________________________
______  ________      _________________________________________________
______  ________      _________________________________________________
______  ________      _________________________________________________
______  ________      _________________________________________________
______  ________      _________________________________________________
______  ________      _________________________________________________
```

Figure 233. SOAP Server Definitions Panel

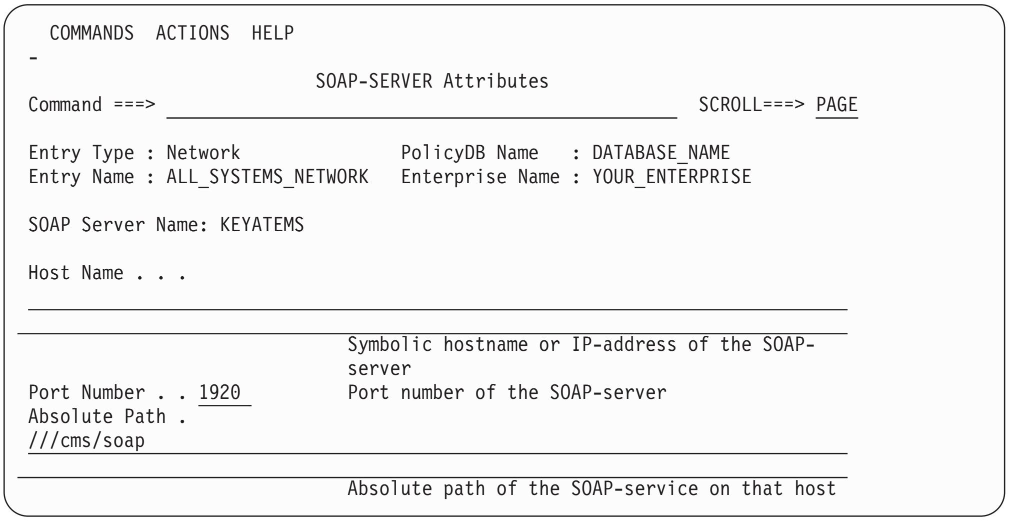
Use this panel to manage your IBM Tivoli Monitoring (ITM) SOAP server definitions.

Action
Use this field to select (S) a server for further processing or to delete (D) a server.

Server Name
Enter a unique 8-character name for the server. Any alphabetic, numeric, #, @ and $ character can be used. The first character must be alphabetic.

Description
Enter descriptive text for the server.

Selecting a server leads to the SOAP-SERVER Attributes panel, as shown in Figure 234 on page 307.



```
  COMMANDS  ACTIONS  HELP
-
                          SOAP-SERVER Attributes
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Network              PolicyDB Name   : DATABASE_NAME
Entry Name : ALL_SYSTEMS_NETWORK  Enterprise Name : YOUR_ENTERPRISE

SOAP Server Name: KEYATEMS

Host Name . . .
_____________________________________________________________________________

_____________________________________________________________________________
                              Symbolic hostname or IP-address of the SOAP-
                              server
Port Number . . 1920          Port number of the SOAP-server
Absolute Path .
///cms/soap___________________________________________________________________

_____________________________________________________________________________
                              Absolute path of the SOAP-service on that host
```

Figure 234. SOAP-SERVER Attributes Panel

Use this panel to define the SOAP server and its attributes. The following fields can be specified:

Host Name
This is the address of the host where the SOAP server is running. It can be a symbolic host name or an IP address. Both IPv4 and IPv6 address formats are allowed.

An IPv6 address consists of hexadecimal characters separated by one or more colon and can be 2–39 characters long.

An IPv4 address consists of 4 decimal qualifiers each in the range 0–255, separated by a dot.

Symbolic host names consists of alphanumeric characters and dots.

Port Number
This is the port number that the SOAP server is listening to. The default for the ITM SOAP server is 1920.

Absolute Path
This is the absolute path of the SOAP service on the host that completes the address of a SOAP service. If there is no path, specify a forward slash ('/') to denote the root directory as the SOAP-servers path. The default path used for ITM SOAP services is ///cms/soap.

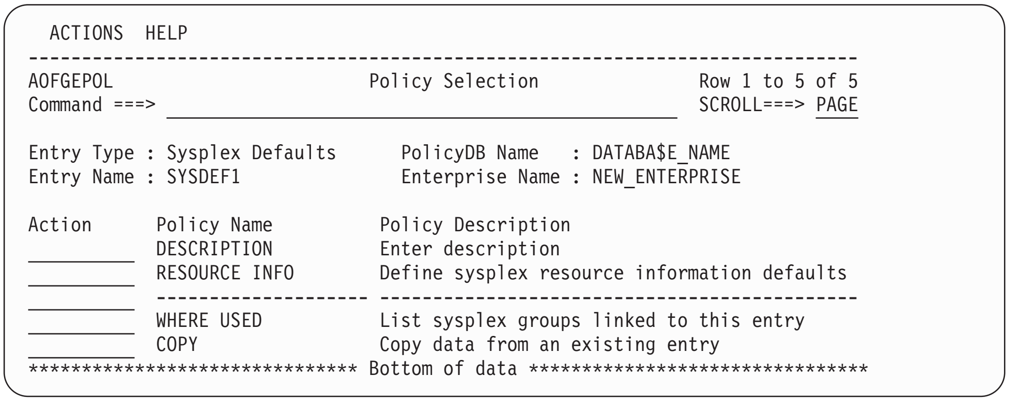
For more details about how SA z/OS uses SOAP messages and how to use them see *IBM Tivoli System Automation for z/OS Customizing and Programming*.

Sysplex Defaults Policy Object

The Sysplex Defaults policy object allows you to define sysplex resource information defaults.

Policy Items for Sysplex Defaults

Figure 235 on page 308 shows the Entry Selection panel to define sysplex resource information defaults.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection              Row 1 to 5 of 5
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : Sysplex Defaults      PolicyDB Name   : DATABA$E_NAME
Entry Name : SYSDEF1               Enterprise Name : NEW_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  RESOURCE INFO        Define sysplex resource information defaults
__________  -------------------- --------------------------------------------
__________  WHERE USED           List sysplex groups linked to this entry
__________  COPY                 Copy data from an existing entry
******************************* Bottom of data ********************************
```

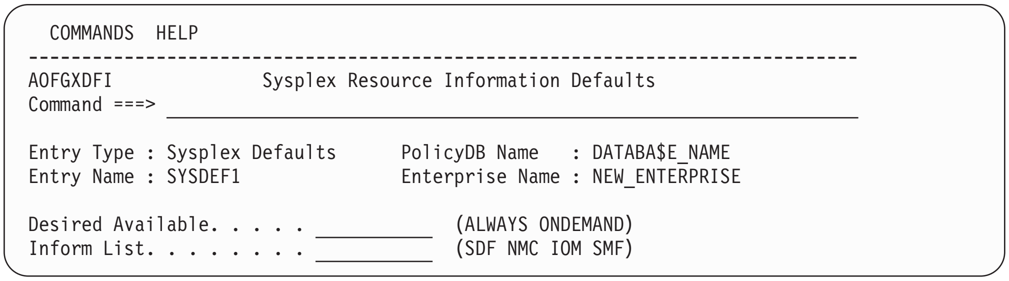
Figure 235. Policy Selection Panel for Sysplex Defaults

The unique policy item for the Sysplex Defaults policy object is:

RESOURCE INFO
Use this policy to specify where sysplex resources are registered to and how the desired status of the resource is determined in the absence of any propagated DesiredStatus votes on the resource.

RESOURCE INFO Policy Item

Figure 236 shows the Sysplex Resource Information Defaults panel that you can use to define where sysplex resources are registered.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGXDFI              Sysplex Resource Information Defaults
Command ===> _________________________________________________________________

Entry Type : Sysplex Defaults      PolicyDB Name   : DATABA$E_NAME
Entry Name : SYSDEF1               Enterprise Name : NEW_ENTERPRISE

Desired Available. . . . . ___________  (ALWAYS ONDEMAND)
Inform List. . . . . . . . ___________  (SDF NMC IOM SMF)
```

Figure 236. Sysplex Resource Information Defaults Panel

Fill in the following field:

Desired Available
Specifies how the desired state of the resource is determined in the absence of any propagated DesiredStatus votes on the resource.

You can specify the following values:

ALWAYS
The resource's desired status is set to AVAILABLE. The resource will be started once its dependencies are satisfied. This is the same behavior that older releases of SA z/OS exhibit.

ONDEMAND
The resource's desired status is set to UNAVAILABLE. The resource will not be started. For SA z/OS to start the resource it must be propagated a MakeAvailable vote.

Note: If nothing is specified the default value is ALWAYS.

Inform List
This field allows you to specify:

- Where the application is registered to
- Where all status changes are propagated to
- Which SMF records are written

Valid receivers are SDF, NMC IOM, and SMF. You can specify more than one receiver. The list of values can be blank or comma separated.

Resident CLISTs Policy Object

> **Note:**
> Since NetView 1.4, NetView automatically loads frequently used CLISTs into storage. You therefore no longer need to define your own resident CLISTs.

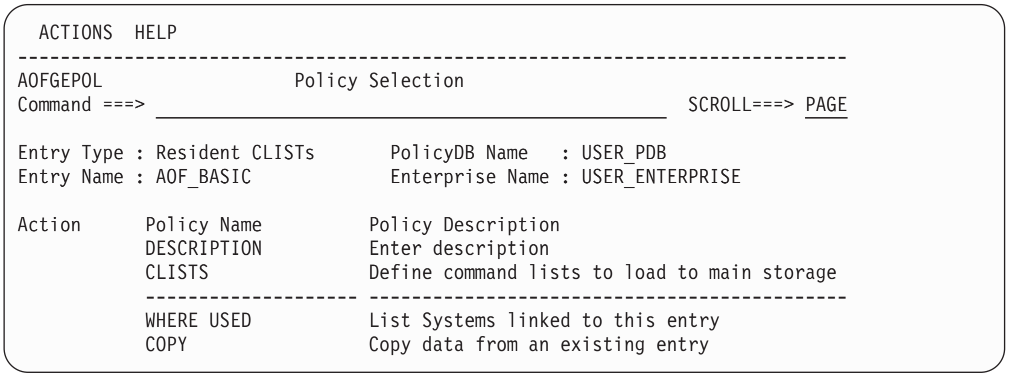
A resident CLIST is a CLIST that is loaded into and kept in NetView storage for the first and all subsequent calls.

Resident CLISTs can improve performance by removing the overhead required to access CLISTs stored on a direct access storage device (DASD) each time the CLIST is called. Resident CLISTs are automatically loaded into main storage, using the NetView LOADCL command, during initialization of the automation environment. Refer to *Tivoli NetView for z/OS Automated Operations Network User's Guide* for more information on using the LOADCL command for CLISTs.

If a new automation control file is loaded after SA z/OS is up and initialized, and that automation control file contains fewer automation procedures in the resident portion than the original automation control file loaded at initialization, even if the procedures are not specified in the new file, they will not be dropped from storage. For example, if the original automation control file contained procedures A, B and C to be loaded resident, and the new automation control file contains procedures A, B, and D to be loaded resident, procedure D will be loaded but procedure C will not be dropped.

Policy Items for Resident CLISTs

Figure 237 shows the policy items that are displayed when a new Resident CLIST entry is defined or when an existing Resident CLIST entry is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                 Policy Selection
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Resident CLISTs       PolicyDB Name   : USER_PDB
Entry Name : AOF_BASIC             Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            DESCRIPTION          Enter description
            CLISTS               Define command lists to load to main storage
            -------------------- --------------------------------------------
            WHERE USED           List Systems linked to this entry
            COPY                 Copy data from an existing entry
```

Figure 237. Policy Selection Panel for Resident CLISTs

The unique policy item for Resident CLIST sets is:

CLISTS
This option allows you to enter the names of the CLISTs you want to be made resident during initialization of SA z/OS.

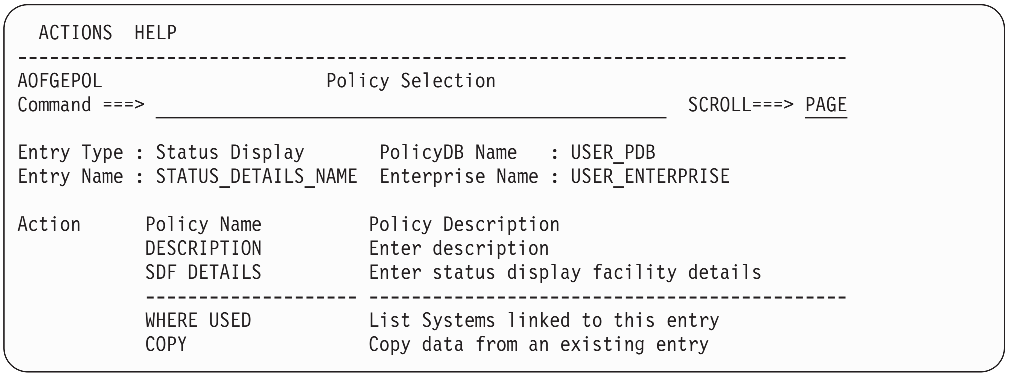
Status Display Policy Object

The status display facility (SDF) provides a display of automated systems and resources using assigned status colors. An operator monitors the status of automated systems and resources by viewing the SDF main panel. Typically, for example, an application shown in green on an SDF status panel indicates the application is up, while red indicates the application is stopped or in a problem state. Operators can use the SDF to monitor the system and decide which actions to take when problems occur. This section describes the entries used to define the SDF in the customization dialog. You only need to change these entries if you use values other than the SA z/OS-provided defaults. The entries specified here define statuses and the priorities assigned to those statuses. These entries are used by SA z/OS common routines to gather data for requests to add status descriptors to status components. Additional parameters manage existing status descriptors and eliminate redundancy.

Note:
Default SDF entries are no longer provided in the sample policy databases because it is not required in SCR definition. These SDF entries are automatically loaded during SA z/OS initialization or an INGAMS REFRESH.

Policy Items for Status Display

Figure 238 shows the policy items that are displayed when a new Status Display policy object is defined or when an existing Status Display object is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                     Policy Selection
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Status Display       PolicyDB Name   : USER_PDB
Entry Name : STATUS_DETAILS_NAME  Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
            DESCRIPTION          Enter description
            SDF DETAILS          Enter status display facility details
            -------------------- --------------------------------------------
            WHERE USED           List Systems linked to this entry
            COPY                 Copy data from an existing entry
```

Figure 238. Policy Selection Panel for Status Display

The unique policy item for Status Display is:

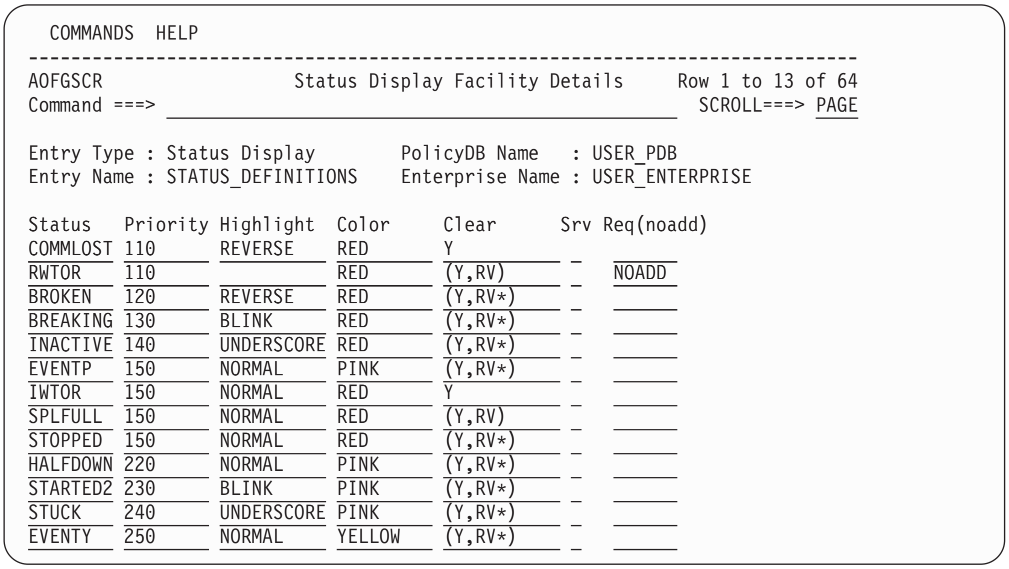
SDF DETAILS
You can use this policy item to define colors and highlights associated with each status.

Refer to “SDF DETAILS Policy Item” on page 311 for more information about this policy item.

For more information on SDF customization, see *IBM Tivoli System Automation for z/OS User's Guide*.

SDF DETAILS Policy Item

A panel like the one in Figure 239 is displayed if you select the SDF DETAILS policy item from the Policy Selection panel for Status Display. The values shown in Figure 239 are the SA z/OS SDF defaults. These entries are shown as an example but are not required in the definition.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGSCR                  Status Display Facility Details     Row 1 to 13 of 64
Command ===>                                                  SCROLL===> PAGE

Entry Type : Status Display        PolicyDB Name   : USER_PDB
Entry Name : STATUS_DEFINITIONS    Enterprise Name : USER_ENTERPRISE

Status   Priority Highlight  Color     Clear      Srv Req(noadd)
COMMLOST 110      REVERSE    RED       Y           _
RWTOR    110                 RED       (Y,RV)      _    NOADD
BROKEN   120      REVERSE    RED       (Y,RV*)     _
BREAKING 130      BLINK      RED       (Y,RV*)     _
INACTIVE 140      UNDERSCORE RED       (Y,RV*)     _
EVENTP   150      NORMAL     PINK      (Y,RV*)     _
IWTOR    150      NORMAL     RED       Y           _
SPLFULL  150      NORMAL     RED       (Y,RV)      _
STOPPED  150      NORMAL     RED       (Y,RV*)     _
HALFDOWN 220      NORMAL     PINK      (Y,RV*)     _
STARTED2 230      BLINK      PINK      (Y,RV*)     _
STUCK    240      UNDERSCORE PINK      (Y,RV*)     _
EVENTY   250      NORMAL     YELLOW    (Y,RV*)     _
```

Figure 239. Status Display Facility Details Panel

The fields are:

Status Any required name. The name must start with an alphabetic or numeric character or one of '@, #, or $'. All other characters must be alphabetic, numeric, or one of '@, #, $, ?, _, !, or ¢. This value can be up to 8 characters long. It is recommended that you use the SA z/OS default statuses.

Priority
The priority number assigned to a particular status type. PRIORITY must be a valid number in the range specified in the AOFINIT member. Normally, priority determines the color displayed for a status. This is the recommended method. If the priority is not defined, the default is 99999999.

Highlight
The type of highlighting used for a particular status type. If this field is not specified, or HL=P is specified, highlighting is determined by the panel definition.

Color The color used for a particular status type. This is the color override entry. Use this parameter to override the priority and color relationship that is defined in the AOFINIT member. Although the color of the status descriptor is changed, the priority is still determined by the order that the status descriptor is chained in.

There is no default color associated with an SA z/OS status. Any color specifications defined using this parameter override the color defined for the range that the priority is in.

Clear The CLEAR parameter helps manage redundant status descriptors. With

SA z/OS, the reference value on a status descriptor is usually the status component name. This option is only available for statuses defined in the automation control file and accessed by SA z/OS common routines described in *IBM Tivoli System Automation for z/OS Programmer's Reference*. The following values are allowed:

- CLEAR=Y

  Deletes any existing status descriptor for this status component with the same reference value as the status descriptor being added. This verifies that, for any given status component, there is only one status descriptor reflecting the true status.
- CLEAR=(Y,RV*)

  Performs the same function as CLEAR=Y, and also deletes status descriptors with reference values prefixed with the same reference value (followed by anything else). For example, if the existing status descriptor for RMF has a reference value of RMF1, and the status descriptor being added has a reference value of RMF, the RMF1 status descriptor is deleted before the new RMF status descriptor is added.
- CLEAR=(Y,RV)

  Performs the same function as CLEAR=(Y,RV*) but the reference value must be the same (not a wildcard).
- CLEAR=(Y,RV*,DA)

  Performs the same function as CLEAR=(Y,RV*), and also deletes all status descriptors with similar data in the data field. (The data field is defined with the DATA keyword on the SDF request that adds the status descriptor to a chain of status descriptors.)
- CLEAR=(Y,RV2*,CO)

  Performs the same function as CLEAR=(Y,RV*) with the following differences:

  1. The color of the existing status descriptor must be the same as that specified in the color entry before it is deleted
  2. Status descriptors having reference values prefixed with the new reference value followed by the number 2 are deleted, provided the color matches the status descriptor being added.
- CLEAR=N

  Indicates that redundant status descriptors are not deleted.

Srv Determines the format of the reference value generated for this entry. If Y is specified for this entry, the generated reference value has the following format:

```
Status_component/Status
```

Req (no add)
Checks the SDF request. REQ=NOADD means the request is not added.

Notes:

1. Choosing reference values carefully simplifies how SDF handles adding and deleting status descriptors.
2. Always use the status component either as the reference value or as a prefix to a reference value.

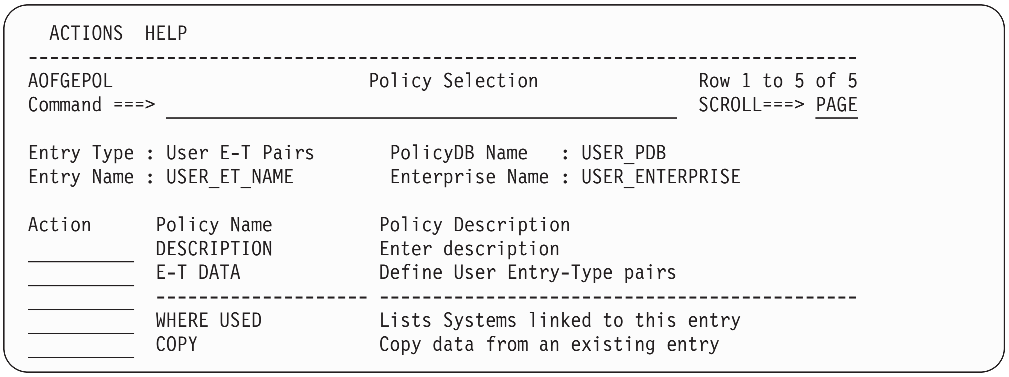
User E-T Pairs Policy Object

This policy object allows you to create your own entry-type pairs and associated keywords in the same format as those used by SA z/OS.

You can use User E-T Pairs to store data for your own automation procedures in the automation control file. User E-T Pairs are provided for situations where there is no suitable, predefined SA z/OS policy object. Using the automation control file to store policy-specific information helps keep your automation procedures generic.

Policy Items for User E-T Pairs

Figure 240 shows the policy items that are displayed when a new User E-T Pairs policy object is defined or when an existing User E-T Pairs object is selected from the Entry Name Selection panel.



```
  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGEPOL                        Policy Selection              Row 1 to 5 of 5
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : User E-T Pairs       PolicyDB Name   : USER_PDB
Entry Name : USER_ET_NAME         Enterprise Name : USER_ENTERPRISE

Action      Policy Name          Policy Description
__________  DESCRIPTION          Enter description
__________  E-T DATA             Define User Entry-Type pairs
__________  -------------------- ---------------------------------------------
__________  WHERE USED           Lists Systems linked to this entry
__________  COPY                 Copy data from an existing entry
```

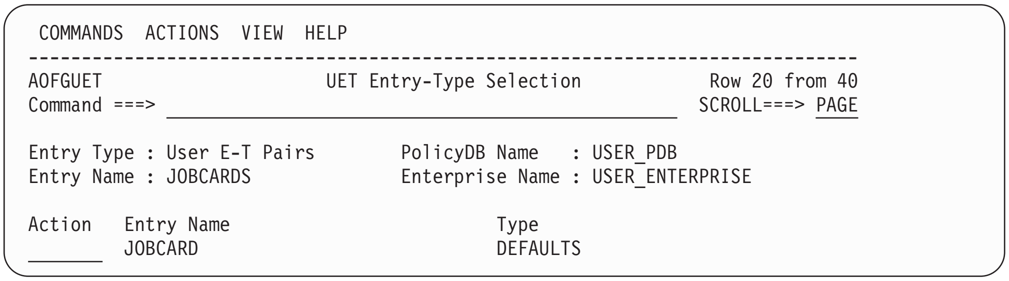
Figure 240. Policy Selection Panel for User E-T Pairs

The unique policy item for User E-T Pairs is:

E-T DATA
This policy item enables you to specify entry-type pairs for the set. You can also enter a description for each pair. Refer to "E-T DATA Policy Item" for more information on this policy item.

E-T DATA Policy Item

Figure 241 shows the UET Entry-Type Selection panel.



```
 COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGUET                     UET Entry-Type Selection            Row 20 from 40
Command ===> ________________________________________________ SCROLL===> PAGE

Entry Type : User E-T Pairs        PolicyDB Name   : USER_PDB
Entry Name : JOBCARDS              Enterprise Name : USER_ENTERPRISE

Action   Entry Name                          Type
_______  JOBCARD                             DEFAULTS
```

Figure 241. UET Entry-Type Selection Panel

The **Entry Name** and **Type** fields are used to define a unique set of key-data pairs. In general, the **Entry Name** field can contain any name except subsystem names existing in the policy database. The **Type** field can be used to identify groups of information for the entry name. You can define wild cards and types by appending the asterisk (*) character to the type value.

The following actions are possible:

S(ELECT)
To select the entry-type pair for further processing. Another panel, similar to Figure 242, is displayed where the keyword-data pairs can be entered.

E(DIT)
To select the entry-type pair for changing the value of entry, type, short and long description.

D(ELETE)
To delete the complete entry-type pair with all associated keyword-data pairs.

The following VIEWs are possible:

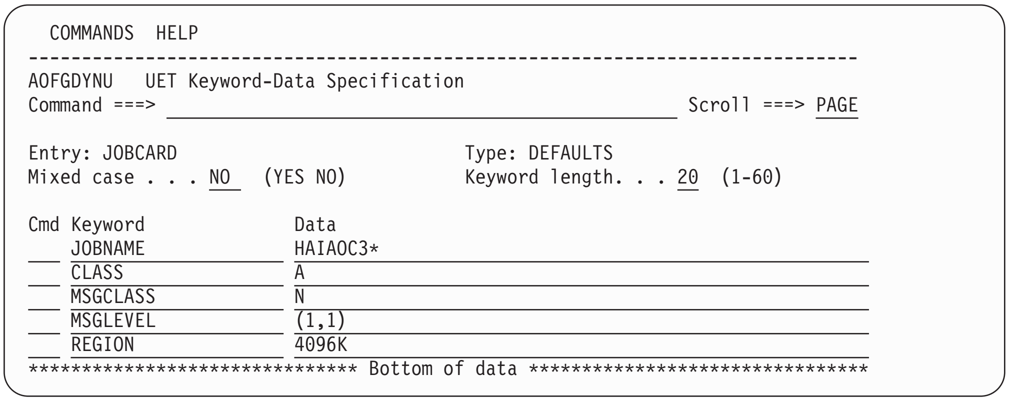
- View entry-type pairs only
- View entry-type pairs with their short description
- View entry-type pairs with their long description

When you enter an E for Edit in the **Action** column, another panel, similar to Figure 242, is displayed where you can enter the keywords and user data for each Entry-Type pair.

A keyword is an identifier for the data that follows it. The set of keyword/data pairs may be null, and each keyword does not have to be unique; that is, each keyword may have multiple sets of data assigned to it.

The following rules apply for the user data:

- Data containing the special characters ()′,″ or imbedded spaces must be enclosed in delimiters. Valid delimiters are single quotation marks, double quotation marks, or parentheses.
- Any imbedded single quotation marks will be automatically doubled by the processing logic of the UET Keyword-Data Specification panel. For example, ABC′DEF′GHI will be changed to ABC″DEF″GHI
- Imbedded single quotation marks must appear in pairs.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGDYNU   UET Keyword-Data Specification
Command ===> ________________________________________________ Scroll ===> PAGE

Entry: JOBCARD                           Type: DEFAULTS
Mixed case . . . NO   (YES NO)           Keyword length. . . 20  (1-60)

Cmd Keyword             Data
___ JOBNAME             HAIAOC3*
___ CLASS               A
___ MSGCLASS            N
___ MSGLEVEL            (1,1)
___ REGION              4096K
******************************* Bottom of data ********************************
```

Figure 242. UET Keyword-Data Specification Panel

The UET Keyword-Data Specification panel allows you to manage the list of keyword-data pairs for one entry-type pair of one User E-T pair entry.

You can define multiple keyword-data pairs for each entry-type pair. By default, the first 20 characters of the keyword value is shown. The length of the data value that is displayed depends on your screen width (for a 24x80 screen, it is 55 characters).

From this panel you can:

- Specify whether the value entered in the **Data** field remains unchanged (Mixed case=YES) or is automatically translated to uppercase characters (Mixed case=NO)
- Change the length of the field that the keyword value is displayed in using the **Keyword length** field
- Use the following line commands on the existing lines to enable you to easily add, remove and manage keyword-data pairs:

  I or I*nn*
  Insert one or *nn* new lines after the selected line

  R or R*nn*
  Repeat the selected line one or *nn* times

  D or D*nn*
  Delete the selected line and the following *nn* lines

  M or M*nn*
  Move the selected line and the following *nn* lines

  MM Move a block of lines

  A Move lines after the selected line

  B Move lines before the selected line
- Change data by overtyping the current values.

If necessary, use F11 to scroll to the right to view more of the data field.

From within a CLIST, you can retrieve the data entered in the UET Keyword-Data Specification panels using the ACFFQRY command. An example is provided in "How to Use User E-T Pairs" on page 83

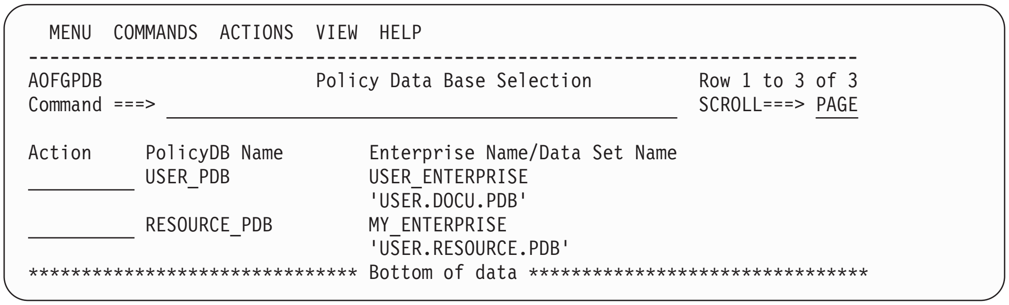
More than one set of User E-T Pairs may be attached to a system, but it is recommended that you attach only one to avoid data being overwritten during initialization of SA z/OS.

Chapter 6. Policy Database Maintenance

Option **4 (Policies)** of the Customization Dialog Primary Menu panel allows you to maintain the list of your policy databases and to access various panels for selected policy databases.

Policy Database Selection List

If you have already created one or more policy databases, they are listed in the Policy Database Selection panel, as shown in Figure 243.



```
  MENU  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGPDB                    Policy Data Base Selection         Row 1 to 3 of 3
Command ===> ________________________________________________ SCROLL===> PAGE

Action     PolicyDB Name       Enterprise Name/Data Set Name
__________ USER_PDB            USER_ENTERPRISE
                               'USER.DOCU.PDB'
__________ RESOURCE_PDB        MY_ENTERPRISE
                               'USER.RESOURCE.PDB'
****************************** Bottom of data ********************************
```

Figure 243. Policy Database Selection Panel

Databases created with the NEW command automatically appear in your policy database list. In addition you can add other databases. To add a database, you need to know the name of its data set and have the appropriate authority to update it. You can work with one policy database from your list at a time.

To select a policy database:

1. Move the cursor to the line containing the database you want to use.
2. Enter s in the **Action** column on that line.

The selected policy database is opened, and the Entry Type Selection panel for that database is displayed. At this point you are ready to start defining your enterprise.

The commands and actions available from the Policy Database Selection panel are:

(O) OPEN
This option lets you access the policy data for that database. The Entry Type Selection panel is displayed. With this option, you can edit the contents of policy database . See Figure 2 on page 7.

(N) NEW
This option lets you create a new policy database as described in “Creating a New Policy Database” on page 319. The Create a New Policy Database panel is displayed. See “Creating a New Policy Database” on page 319 for more details.

(E) EDIT
This option lets you change the Enterprise name and data set name of your policy database for the build, report, and log output. To change the data within the database, you must OPEN it. Using the EDIT option, you can also start and stop the logging of user activity. For more information, see “Specifying Settings for Policy Databases” on page 324.

(D) DELETE

This option lets you delete the policy database data set, and all references to the policy database. SA z/OS will display a panel asking you to confirm your request.

SA z/OS does not provide any extended security to prevent you from deleting policy database data sets. This must be provided by a System Authorization Facility (SAF) product such as RACF®.

Note:

SA z/OS does not allow you to delete more than one policy database at a time. If you enter multiple delete requests in the **Action** column, the first request is processed and the remainder are ignored.

(A) ADD

This option lets you add a policy database to the displayed list. You enter a policy database name, an enterprise name and the name of a policy database data set that already exists.

(M) REMOVE

This option lets you remove the policy database details from the displayed list. The data set itself is not affected.

Note:

If you remove a policy database from *all* display lists, the policy database still exists as a z/OS data set.

(B) BUILD

This option displays the Build Functions Menu. You can build various configuration files. This function is explained in Chapter 8, “Building and Distributing Configuration Files,” on page 337.

(G) MANAGE

This option will take you to the Data Management Menu panel shown in Figure 261 on page 351, which allows you to import policies or migrate an automation control file to a policy database of the current release of SA z/OS.

You can find more information about this option in Chapter 9, “Data Management,” on page 351.

(R) REPORT

This option displays the Report Selection Menu for the selected policy database. You can create a report of all automation data in the policy database. The report offers options to report either the full automation policy, or to select a specific entry type and name to be reported.

You can either create an unformatted file to be viewed in a TSO session, or an HTML file to be viewed by an Internet Browser.

The following reports can be created:

- A report for all objects in the policy database, that will not be used for building the system operations configuration files (unlinked entries and relationships).
- A report for all automatically generated resources, which are the resources of type APL, APG, and MTR.

- A report of all members in the system operations build output data set, that are not referenced by the actual configuration.
- A status message report.

For more information, see Chapter 7, “Creating Reports about Policy Databases,” on page 329.

(V) VIEW
This option toggles between two views. The summary view displays the policy database name and enterprise name for each policy database on one line. The extended view displays a second line for each policy database, giving the data set name.

(L) LOCATE
Entering LOCATE lets you position yourself within the displayed list. If the entry is identical, the selected entry is the first in the list. If the entry is not identical, the next higher (alphabetically) entry is the second in the list. This is the same as the ISPF LOCATE function.

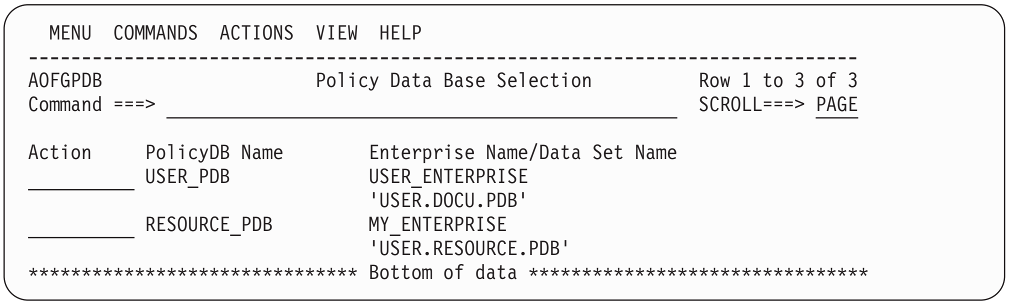
Creating a New Policy Database

A policy database is a partitioned data set (PDS) that contains policy data in its members. To create a new policy database, follow these steps:

Step 1. Select option 4 *(Policies)* on the Customization Dialog Primary Menu .

This takes you to the Policy Database Selection panel, shown in Figure 244, which lists all policy databases that are currently available.

You can toggle the display of the data set and enterprise name in the VIEW pull-down menu.

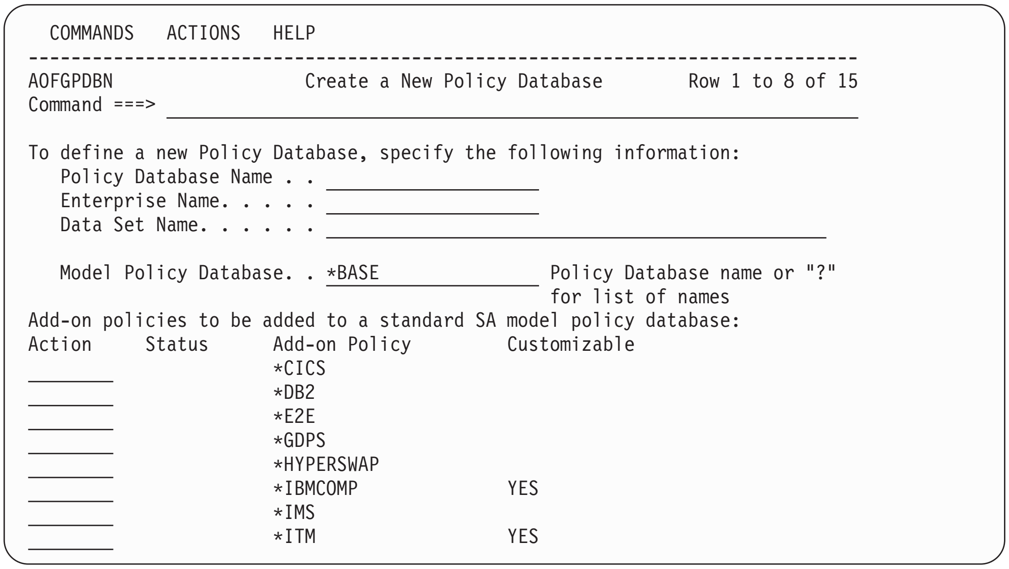


```
  MENU  COMMANDS  ACTIONS  VIEW  HELP
------------------------------------------------------------------------------
AOFGPDB                    Policy Data Base Selection         Row 1 to 3 of 3
Command ===> ________________________________________________ SCROLL===> PAGE

Action     PolicyDB Name       Enterprise Name/Data Set Name
_________  USER_PDB            USER_ENTERPRISE
                               'USER.DOCU.PDB'
_________  RESOURCE_PDB        MY_ENTERPRISE
                               'USER.RESOURCE.PDB'
****************************** Bottom of data ********************************
```

Figure 244. Policy Database Selection Panel

Step 2. Enter `N` or `NEW` on the command line.

This takes you to the Create a New Policy Database panel, shown in Figure 245 on page 320.



```
  COMMANDS   ACTIONS   HELP
------------------------------------------------------------------------------
AOFGPDBN                 Create a New Policy Database       Row 1 to 8 of 15
Command ===> _________________________________________________________________

To define a new Policy Database, specify the following information:
   Policy Database Name . . ____________________
   Enterprise Name. . . . . ____________________
   Data Set Name. . . . . . _______________________________________________

   Model Policy Database. . *BASE______________ Policy Database name or "?"
                                                for list of names
Add-on policies to be added to a standard SA model policy database:
Action     Status      Add-on Policy        Customizable
                       *CICS
________               *DB2
________               *E2E
________               *GDPS
________               *HYPERSWAP
________               *IBMCOMP             YES
________               *IMS
________               *ITM                 YES
________
```

Figure 245. Create a New Policy Database Panel

Step 3. Fill in the fields on this panel:

- The **PolicyDB Name** field is mandatory. Enter the name of the new policy database; this can be up to twenty characters in length.
- The **Enterprise Name** field is mandatory. This can be up to twenty characters in length. You can use the same value as in the **PolicyDB Name** field.
- The **Data Set Name** field is mandatory. This can be any valid z/OS data set name. Standard rules for quoted and unquoted data set names apply.

  If you enter the name of an existing data set, a warning is issued. If you enter a name that has not been given to an existing data set, a new data set will be created. The policy database that you are creating will be stored in that new data set.
- The **Model PolicyDB name** field is mandatory. This must be an existing policy database name. The new policy database that you are creating will contain a *copy* of the model policy database.

  You can use either of the SA z/OS sample policy databases `*BASE` or `*EMPTY` or a database that is currently in your list of policy databases.

  If you do not want your new policy database to contain data that is copied from an existing policy database, use `*EMPTY` in this field.

  If you want to see what model databases are available, type "?" in this field and press Enter. This will display the Select Model Policy Database panel that you can select a database from.

Note

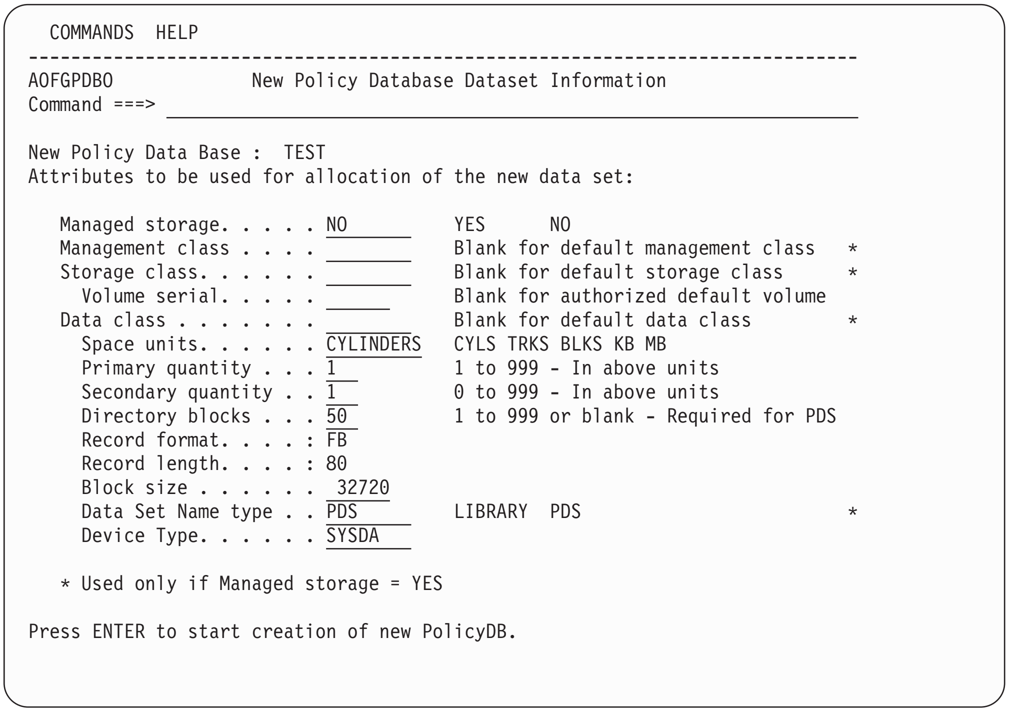
If a user PDB is selected (that is, one without a '*' as the first character), no add-on PDBs can be selected. An error message is issued and pressing Enter does not lead to the next panel.

- Selecting one or more of the sample add-on policies from the list on the panel is optional. If you do make a selection, the new policy database that you are creating will contain a copy of each add-on that you have selected. If an add-on policy is marked as customizable, this means that it contains components that can be separately selected (for example, *ITM contains definitions for OMEGAMON II and OMEGAMON XE monitors).

> **Note:**
> Each of the sample add-on policies that are shipped with SA z/OS contain definitions that fully complement those in *BASE and *EMPTY.

Step 4. Press Enter to submit your input.

This takes you to the New Policy Database Data Set Information panel, shown in Figure 246.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGPDBO            New Policy Database Dataset Information
Command ===> _________________________________________________________________

New Policy Data Base :  TEST
Attributes to be used for allocation of the new data set:

   Managed storage. . . . . NO           YES      NO
   Management class . . . . ________     Blank for default management class   *
   Storage class. . . . . . ________     Blank for default storage class      *
     Volume serial. . . . . ______       Blank for authorized default volume
   Data class . . . . . . . ________     Blank for default data class         *
     Space units. . . . . . CYLINDERS    CYLS TRKS BLKS KB MB
     Primary quantity . . . 1            1 to 999 - In above units
     Secondary quantity . . 1            0 to 999 - In above units
     Directory blocks . . . 50           1 to 999 or blank - Required for PDS
     Record format. . . . : FB
     Record length. . . . : 80
     Block size . . . . . .  32720
     Data Set Name type . . PDS          LIBRARY  PDS                         *
     Device Type. . . . . . SYSDA

   * Used only if Managed storage = YES

Press ENTER to start creation of new PolicyDB.
```

Figure 246. New Policy Database Data Set Information Panel

Step 5. The fields that you must enter data in are listed below. For those fields that are not explicitly explained, refer to the appropriate z/OS literature.

Managed Storage
This indicates whether the data set should be allocated on managed storage. If this field is set to YES, the fields marked with an * are used to allocate the data set. If NO is specified then these fields may be specified but are not used.

Management class
This specifies the management class that should be used to obtain the data management related information for the allocation of the data set.

Storage class
This specifies the storage class that should be used to obtain the storage related information for the allocation of the data set. The following data, related to storage class, can be specified if you are

not using managed storage or you have not specified a storage class or if you wish to override the default storage class values.

Volume serial
The volume that the database is to be stored on. If this field is left blank, it defaults to your local system default.

Data class
This specifies the data class that should be used to obtain the data related information for the allocation of the data set. The following data, related to data class, can be specified if you are not using managed storage or you have not specified a data class or if you wish to override the default data class values:

Space Units
Valid allocation units for the data set are:
- CYLINDERS (this is the default)
- TRACKS
- BLOCKS
- MEGABYTES
- KILOBYTES

Primary quantity
This specifies the primary space allocation in space units. The default is 1.

Secondary quantity
This specifies the secondary space allocation in space units. The default is 1.

Directory blocks
Number of blocks to allocate for the directory. This field is only used when allocating a PDS. The default is 50.

Record format
This is FB and cannot be changed.

Record length
This is 80 and cannot be changed.

Block size
This defines the block size for the data set. It must be a multiple of 80. The default is 32720.

Data Set Name type
This indicates whether the data set is to be allocated as a partitioned data set (PDS) or partitioned data set extended (LIBRARY).

Device Type
The type of storage device. This field defaults to SYSDA.

Note:
Many installations regularly delete data sets created on SYSDA volumes. If you want to create a permanent policy database you should not create it on a SYSDA volume. If the data set for a policy database is deleted, you will receive an error message when you attempt to open it. You can either recreate the data set if you have a backup, or enter M (REMOVE) against the policy database to remove it from your policy database list. Entering D (DELETE) will result in an error, since the data set has already been deleted.

Note:
The DASD allocation figures shown in Figure 246 on page 321 are a suggested minimum only. Your actual storage requirements depend on the size of your enterprise.

Step 6. Press Enter to submit your input.

This creates the new policy database. SA z/OS displays a Command Progress Display panel. When database creation is finished, the Entry Type Selection panel is displayed for the new database.

Using SA z/OS Sample Policies

SA z/OS provides sample enterprise organizations implemented as SA z/OS policies. The sample policies that are shipped with SA z/OS are installed when the SA z/OS customization dialog is installed.

To see what sample policies there are, you need to navigate to the Create a New Policy Database panel as described in “Creating a New Policy Database” on page 319. The asterisk character (*) at the start of a policy name on that panel indicates that it is a sample.

Diagrams of the sample policies are also provided as PDF files that are located in the USS installation path. The default for this path is: /usr/lpp/ing/doc/policies/.

You can never modify the sample policies themselves. All you can do is modify *copies* of the samples. To create a copy of a sample policy, follow the steps in “Creating a New Policy Database” on page 319 and select the sample that you want to copy in step 3 on page 320.

For information on how to *import* a sample policy into an existing policy database, see “Importing Sample Add-On Policies” on page 354.

There are two kinds of sample policy: Basic and Add-On.

The basic policies are:

***EMPTY**
This sample is simply an empty policy database.

***BASE**

This sample policy simplifies the exploitation of basic system automation using SA z/OS. This allows users to startup and shutdown a basic z/OS computing environment within a sysplex.

The *BASE policy contains the main z/OS, networking and security components that are required in most client shops:

1. z/OS Link List Look-Aside
2. z/OS Base components
3. z/OS Base applications
4. Networking components
5. Security components
6. SA z/OS components

The computing environment consists of three systems running within a sysplex. This environment is considered to be an appropriate base for high availability solutions, delivered as add-on model policies. Depending on your needs, these models can easily be tailored to less or more than three systems, just by unlinking or removing, or adding or linking systems.

A single system needs to be defined as a sysplex with only one system.

The add-on policies are:

***CICS**

***DB2**

***E2E**

***GDPS**

***HYPERSWAP**

***IBMCOMP**

***IMS**

***ITM**

***NMC**

***PROCOPS**

***SAP**

***TBSM**

***TWS**

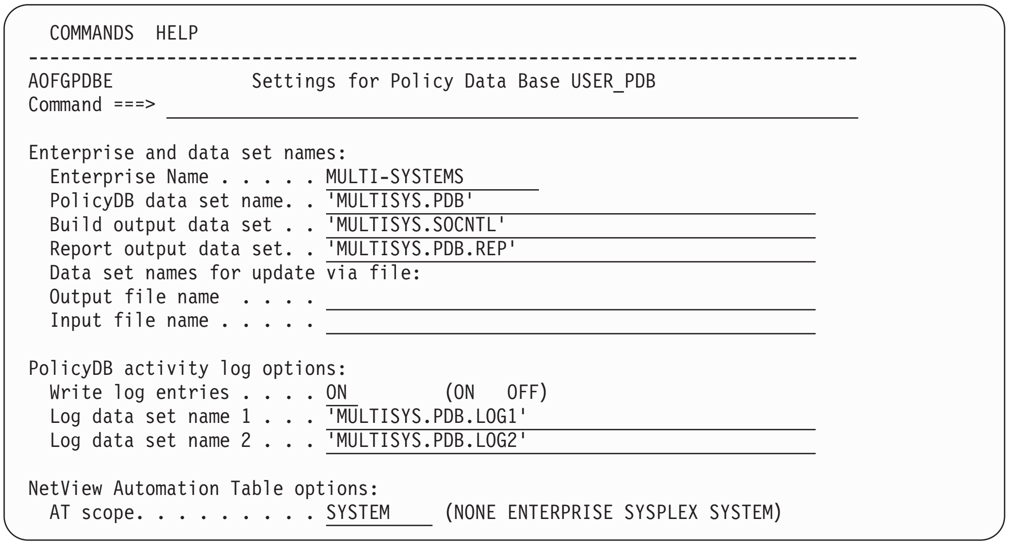
***USS**

***WEBSPHERE**

For information on what is in the sample add-on policies, use the view (v) action on the Create a New Policy Database panel (shown in Figure 245 on page 320).

Specifying Settings for Policy Databases

Specifying the Edit option from the Policy Database Selection panel for a policy database leads to the Settings for Policy Database panel, as shown in Figure 247 on page 325.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGPDBE             Settings for Policy Data Base USER_PDB
Command ===>

Enterprise and data set names:
  Enterprise Name . . . . . MULTI-SYSTEMS
  PolicyDB data set name. . 'MULTISYS.PDB'
  Build output data set . . 'MULTISYS.SOCNTL'
  Report output data set. . 'MULTISYS.PDB.REP'
  Data set names for update via file:
  Output file name  . . . .
  Input file name . . . . .

PolicyDB activity log options:
  Write log entries . . . . ON         (ON   OFF)
  Log data set name 1 . . . 'MULTISYS.PDB.LOG1'
  Log data set name 2 . . . 'MULTISYS.PDB.LOG2'

NetView Automation Table options:
  AT scope. . . . . . . . . SYSTEM     (NONE ENTERPRISE SYSPLEX SYSTEM)
```

Figure 247. Settings for Policy Database Panel

The first set of fields on this panel allows you to specify enterprise and data set names, as follows:

Enterprise Name
This allows you to re-specify the enterprise name. If this policy database has also been added to the Policy Data Base Selection list of other users they need to REMOVE or ADD the policy database to get the new name.

PolicyDB data set name
This allows you to re-specify the enterprise data set name.

Build output data set
This allows you to specify where you want the configuration files that are created by the System Operations build to be stored.

Report output data set
This allows you to specify where you want the output, created by the REPORT function, to be stored. It is not recommended that you store the output as a member of the policy database itself due to record length problems.

If you leave the field empty, SA z/OS allocates a data set with a default name (the policy database data set name extended with .REP as the low-level qualifier) and a default size.

If you do specify a data set name, SA z/OS allocates a data set with that name and a default size.

If you preallocate a data set and specify the name here, SA z/OS uses that data set. For the complete policy database report, it is recommended that you use format VB and a record length of 1000 to avoid truncations.

For other reports, a record length of 132 is usually sufficient (see Figure 249 on page 329 for a list of other reports that are available).

Output file name
This specifies where you want the output created by the file update function to be stored. It needs to be a sequential file. You can leave the output file name field empty. If you do, SA z/OS will allocate a data set with a default name (policy database data set name extended with a low level qualifier of .UPD) and a default size. If you specify a sequential file

name that has not been preallocated, SA z/OS will allocate a data set with that name and a default size. If you preallocate a sequential file and specify the name here, SA z/OS will use that file. It is recommended to use format VB and a record length of 255 to avoid truncations.

Input file name
This specifies where you want the input created by the file update function to be stored, or the input for the policy database update to be read from. It needs to be a sequential file.

The second set of fields allows you to set logging options for this policy database. See “Logging Policy Database Modifications” for more details.

The last field on this panel, **AT Scope**, allows you to define the scope of a NetView Automation Table. Valid definitions are:

NONE
No NetView automation table is built.

ENTERPRISE
One NetView automation table is built for the entire policy database.

SYSPLEX
One NetView automation table is built per Sysplex Group.

No NetView automation table is built for systems that are not linked to a sysplex group.

Note:
When the **AT Scope** is set to SYSPLEX, it is strongly recommended that you link all systems contained in the policy database to a group (GRP) of type ’SYSPLEX’.

If this is not done, no AT is built for systems that are not linked to groups at all, or that are linked to groups of type STANDARD. A corresponding warning is issued at configuration build time.

SYSTEM
One NetView automation table is built per system (default). If the AT scope is changed to any value except NONE a build of type ALL will be performed.

Logging Policy Database Modifications

With SA z/OS you can write a log entry whenever a modification has been made for a specific Policy Database by using the policy database activity log option on the Settings for Policy Data Base panel (see Figure 247 on page 325). Fill in the following fields:

Write log entries
Turns logging ON or OFF for this Policy Database. The default value is OFF.

Log data set name
This allows you to specify two data sets for the activity log. Whenever a data set runs out of space or cannot be accessed for some reason, the dialog switches automatically to the alternate data set and continues writing the log records there.

The alternate log data set is cleared and exit routine INGEX09 is called to allow user-specific actions, such as backing up the data set.

If you leave this field blank, SA z/OS allocates data sets with a default name (the policy database data set name extended with .LOG1 or .LOG2 as low level qualifier) and a default size.

If you specify a data set name, SA z/OS allocates a data set with this name and a default size.

If you preallocate a data set and specify the name here, SA z/OS uses this data set.

It is recommended that you use format VB and a record length of 250 to avoid truncations.

If the Settings for Policy Data Base panel is invoked and the specified policy database data set cannot be accessed, the log option fields are initially empty.

Whenever the policy database data set name is changed, the current log options are read from the policy database and displayed on the panel.

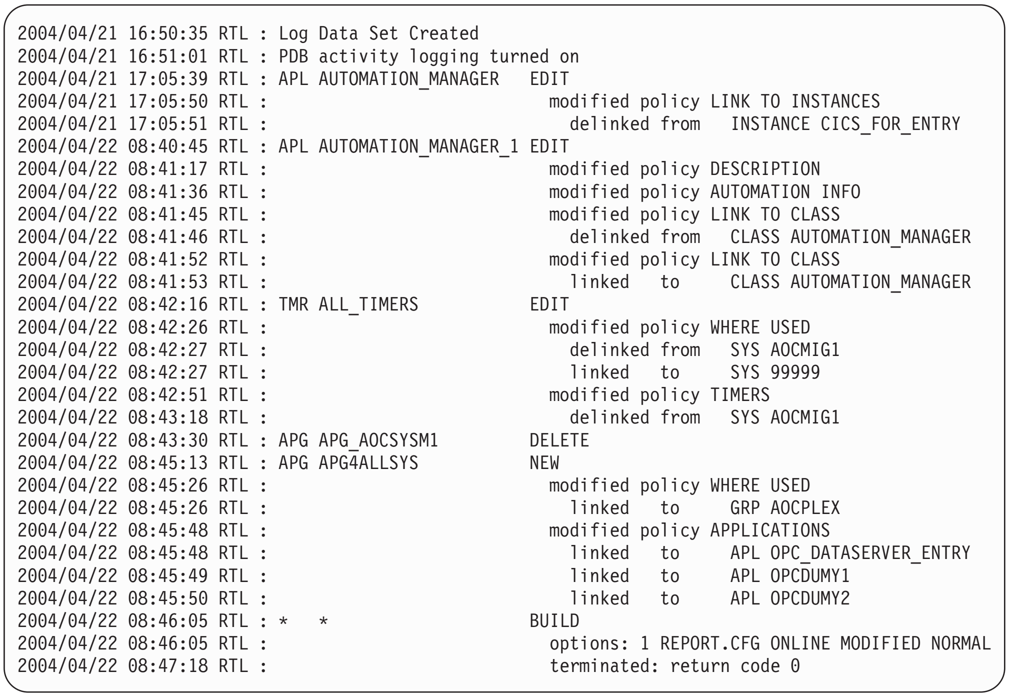
The following information is logged:

- Editing a policy item (for example, changing an automation flag or a command)
- Changing a link, which includes:
  - Entry type links, for example, a link between an application and an application group
  - Links of instances and classes
  - Resource links
- Adding events to a trigger
- Invoking the DATA MANAGEMENT functions (these commands change the policy database content)
- Invoking control file build (it writes build statistics into the policy database)
- Creating and deleting objects
- Copying objects
- Errors (as long as a policy database is open)
- Renaming objects

The following information is *not* logged:

- Option changes (using selection *0 Settings*)
- Data set name changes (EDIT)
- REPORT function invocations
- VIEW functions (for example, view build report)
- Initial conversion of the policy database (when invoked the first time with SA z/OS)
- Invocation of batch functions, for example, batch build

A log entry includes the user ID, timestamp, and the policy item that was changed, or the command name and parameters if available. An example of an activity log is shown in Figure 248 on page 328.

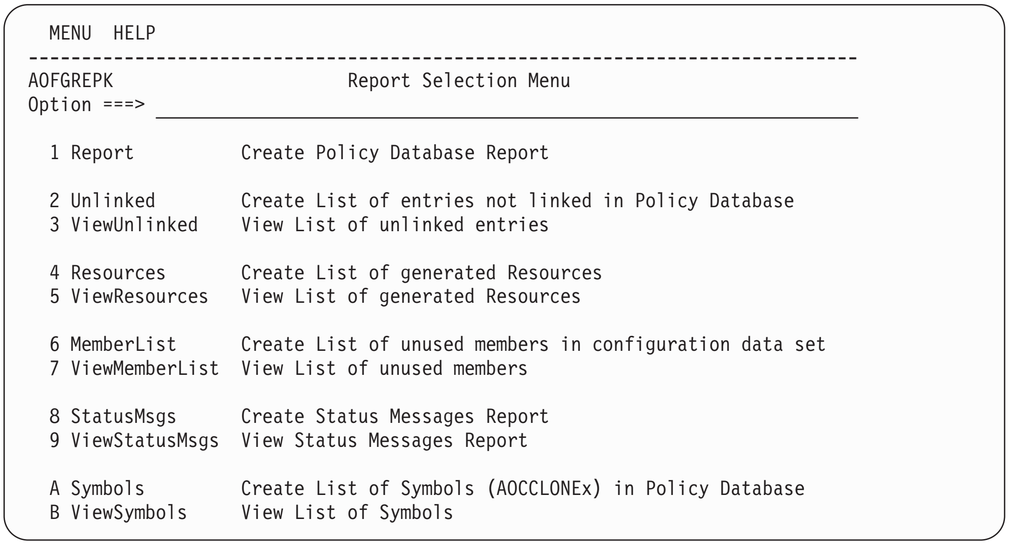


```
2004/04/21 16:50:35 RTL : Log Data Set Created
2004/04/21 16:51:01 RTL : PDB activity logging turned on
2004/04/21 17:05:39 RTL : APL AUTOMATION_MANAGER   EDIT
2004/04/21 17:05:50 RTL :                            modified policy LINK TO INSTANCES
2004/04/21 17:05:51 RTL :                              delinked from   INSTANCE CICS_FOR_ENTRY
2004/04/22 08:40:45 RTL : APL AUTOMATION_MANAGER_1 EDIT
2004/04/22 08:41:17 RTL :                            modified policy DESCRIPTION
2004/04/22 08:41:36 RTL :                            modified policy AUTOMATION INFO
2004/04/22 08:41:45 RTL :                            modified policy LINK TO CLASS
2004/04/22 08:41:46 RTL :                              delinked from   CLASS AUTOMATION_MANAGER
2004/04/22 08:41:52 RTL :                            modified policy LINK TO CLASS
2004/04/22 08:41:53 RTL :                              linked   to     CLASS AUTOMATION_MANAGER
2004/04/22 08:42:16 RTL : TMR ALL_TIMERS           EDIT
2004/04/22 08:42:26 RTL :                            modified policy WHERE USED
2004/04/22 08:42:27 RTL :                              delinked from   SYS AOCMIG1
2004/04/22 08:42:27 RTL :                              linked   to     SYS 99999
2004/04/22 08:42:51 RTL :                            modified policy TIMERS
2004/04/22 08:43:18 RTL :                              delinked from   SYS AOCMIG1
2004/04/22 08:43:30 RTL : APG APG_AOCSYSM1         DELETE
2004/04/22 08:45:13 RTL : APG APG4ALLSYS           NEW
2004/04/22 08:45:26 RTL :                            modified policy WHERE USED
2004/04/22 08:45:26 RTL :                              linked   to     GRP AOCPLEX
2004/04/22 08:45:48 RTL :                            modified policy APPLICATIONS
2004/04/22 08:45:48 RTL :                              linked   to     APL OPC_DATASERVER_ENTRY
2004/04/22 08:45:49 RTL :                              linked   to     APL OPCDUMY1
2004/04/22 08:45:50 RTL :                              linked   to     APL OPCDUMY2
2004/04/22 08:46:05 RTL : *   *                    BUILD
2004/04/22 08:46:05 RTL :                            options: 1 REPORT.CFG ONLINE MODIFIED NORMAL
2004/04/22 08:47:18 RTL :                            terminated: return code 0
```

Figure 248. Example of an Activity Log

Chapter 7. Creating Reports about Policy Databases

If you select option *3 Report* of the Customization Dialog Primary Menu shown in "How to Start the Customization Dialog" on page 34, the panel shown in Figure 249 is displayed, where you can create reports about various SA z/OS objects.

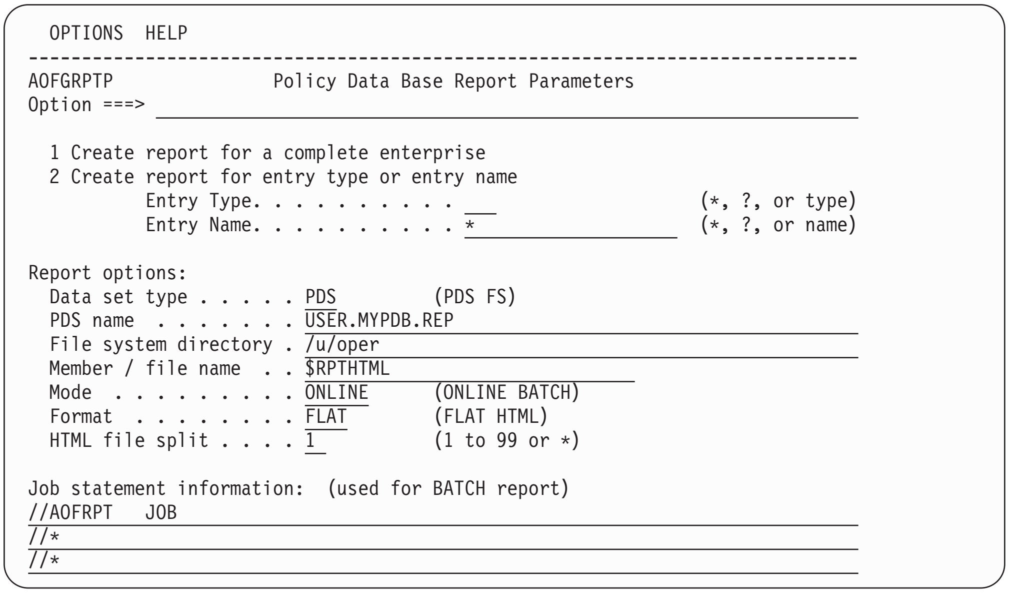


```
  MENU  HELP
------------------------------------------------------------------------------
AOFGREPK                      Report Selection Menu
Option ===> __________________________________________________________________

  1 Report          Create Policy Database Report

  2 Unlinked        Create List of entries not linked in Policy Database
  3 ViewUnlinked    View List of unlinked entries

  4 Resources       Create List of generated Resources
  5 ViewResources   View List of generated Resources

  6 MemberList      Create List of unused members in configuration data set
  7 ViewMemberList  View List of unused members

  8 StatusMsgs      Create Status Messages Report
  9 ViewStatusMsgs  View Status Messages Report

  A Symbols         Create List of Symbols (AOCCLONEx) in Policy Database
  B ViewSymbols     View List of Symbols
```

Figure 249. SA z/OS Report Selection Menu

1 Report

Use this option to create a report that includes all the automation data of the selected policy database. SA z/OS displays the Policy Data Base Report Parameters panel, as shown in Figure 250 on page 330, where you can choose to create either the full automation policy report, or to create a report about a specific entry type and name. You can also run this function in batch mode.



```
  OPTIONS  HELP
------------------------------------------------------------------------------
AOFGRPTP              Policy Data Base Report Parameters
Option ===> __________________________________________________________________

  1 Create report for a complete enterprise
  2 Create report for entry type or entry name
           Entry Type. . . . . . . . . . ___                    (*, ?, or type)
           Entry Name. . . . . . . . . . *____________________  (*, ?, or name)

Report options:
  Data set type . . . . . PDS         (PDS FS)
  PDS name  . . . . . . . USER.MYPDB.REP______________________________________
  File system directory . /u/oper_____________________________________________
  Member / file name  . . $RPTHTML________________________
  Mode  . . . . . . . . . ONLINE      (ONLINE BATCH)
  Format  . . . . . . . . FLAT        (FLAT HTML)
  HTML file split . . . . 1_          (1 to 99 or *)

Job statement information:  (used for BATCH report)
//AOFRPT   JOB________________________________________________________________
//*___________________________________________________________________________
//*___________________________________________________________________________
```

Figure 250. Policy Data Base Report Parameters Panel

Use the **Report options** section of the panel to specify how the report is created and saved.

Data set type
Specify the output type for the report:

PDS
The report is stored as a partitioned data set.

FS The report is written to a UNIX file system (HFS or ZFS).

The default is PDS.

PDS name
If the data set type is PDS, this field can remain empty. A default data set name is assumed and a default allocation is performed. Any user input is translated to upper case.

File system directory
If the data set type is FS, you need to enter the file system directory. There is no default. No uppercase translation is performed. If you specify a directory that does not exist, it is created.

Member / file name
Specify the member or file name for the report. The default is $RPTHTML or $RPTFLAT. Any name that you specify is saved across sessions.

For a PDS it must be a valid TSO member name. When it is downloaded to a PC, you must add the file extension `.html`. Note also that the file name for FS is *case-sensitive*.

If you choose to split an HTML file that is targeted to a PDS, the last 1 or 2 characters of the file name are replaced with a serial number. This is 1–*n* if you specify a single figure split value, or 01–*nn* for a 2-digit value. Thus, $RPTHTML becomes either $RPTHTM1–$RPTHTM*n* or $RPTHT01–$RPTHT*nn*.

If split files are directed to an FS director, the running number is appended to the specified member name, and a .html suffix is added as the files are written.

If a member with the specified or generated split-file name already exists, you are asked to confirm the report generation before it is overwritten.

Mode

Specify whether the report should be created in the foreground (ONLINE) or background (BATCH). The default is ONLINE.

Format

Specify the format of the report:

- FLAT creates a plain text file to be viewed in a TSO session.
- HTML creates an HTML file that you can view with your Internet browser. This version of the report creates hyperlinks for all entry names and resource names that take you to the start of the data for the selected entry or resource. These links also work if the HTML report has been split into several files. Note however that a hyperlink is not created if the application or application group name is not unique (for example, there are identical subsystem names for different applications) or it does not exist in the policy database.

The default is FLAT.

HTML file split

If you have large policy databases that produce large reports you may find that your Internet browser has difficulties handling a single large file. You can use this field to have the HTML report split into several smaller files, as follows:

1 One file is generated. This is the default.

nn **(>1)**

The report is split into *nn* files that are all roughly the same size. The files are split between policy entries.

***** The report output is split with a default size of 10.000 lines per file.

The Command Progress Display panel informs you about the progress of report generation.

You can run a job for batch report creation. The SINGSAMP sample library contains a sample job in member INGEBRPT. The output data set or file system must exist before you run this job.

The example in Figure 251 on page 332 shows a small part of the data listed in the report output format FLAT for the application JES2.

```
   ⋮
========================================
JES2
========================================
Short description              : JES2 policy definition
Long description               : Defines automation and monitoring characteristics
                                                            for the JES2 job.
Application Type               : JES2
Subsystem Name                 : JES2
MVS job name                   : JES2
Scheduling subsystem           : MSTR
Command Prefix                 : $
Start on IPL                   : YES
Subsystem Startup Parameters   : ,PARM=WARM,REL=710,SYS=&SYSNAME.

========== JES2 Drain Specifications ==========
Drain lines                    : YES
Drain VTAM interface           : YES
Drain printers                 : YES
Force drain lines              : $E
Force drain VTAM interface     : $E
Force drain printers           : $I

========== Application Messages and User Data ==========
| Message id   | Description                                 |
--------------------------------------------------------------
| $HASP095     |                                             |
--------------------------------------------------------------

             | Code 1   | Code 2   | Code 3 | Value Returned |
             -------------------------------------------------
             | *        | $PJ*     |        | STOPPING       |
             | ERROR*   | $K03     |        | STOPPING       |
             | ABEND*   | SA22     |        | STOPPING       |
             | *        | *        |        | ABENDING       |

--------------------------------------------------------------
| $HASP098     |                                             |
--------------------------------------------------------------

             | Pass/Selection | Reply Text |
             -------------------------------
             | 1              | DUMP       |
             | 2              | END        |

--------------------------------------------------------------
| POSTSTART    | Executed after startup has been initiated   |
--------------------------------------------------------------

             | Pass  | Automated Function | Command Text                        |
             -------------------------------------------------------------------
             |       | JESOPER            | MVS F BPXOINIT,RESTART=FORKS        |


  ========== Relationships ==========
| Relationship Type | Supporting Resource | Automation | Chaining | Satisfy condition    |
------------------------------------------------------------------------------------------
| PREPUNAVAILABLE   | VTAM/APL/=          | PASSIVE    | WEAK     | WhenDown             |
   ⋮
```

Figure 251. Example of Report Output Format FLAT

2 Unlinked

Use this option to create a report of all entries within the current policy database that would be ignored during BUILD processing. This includes:

- all objects in the policy database that are not linked to a system or sysplex
- all non-MVS systems

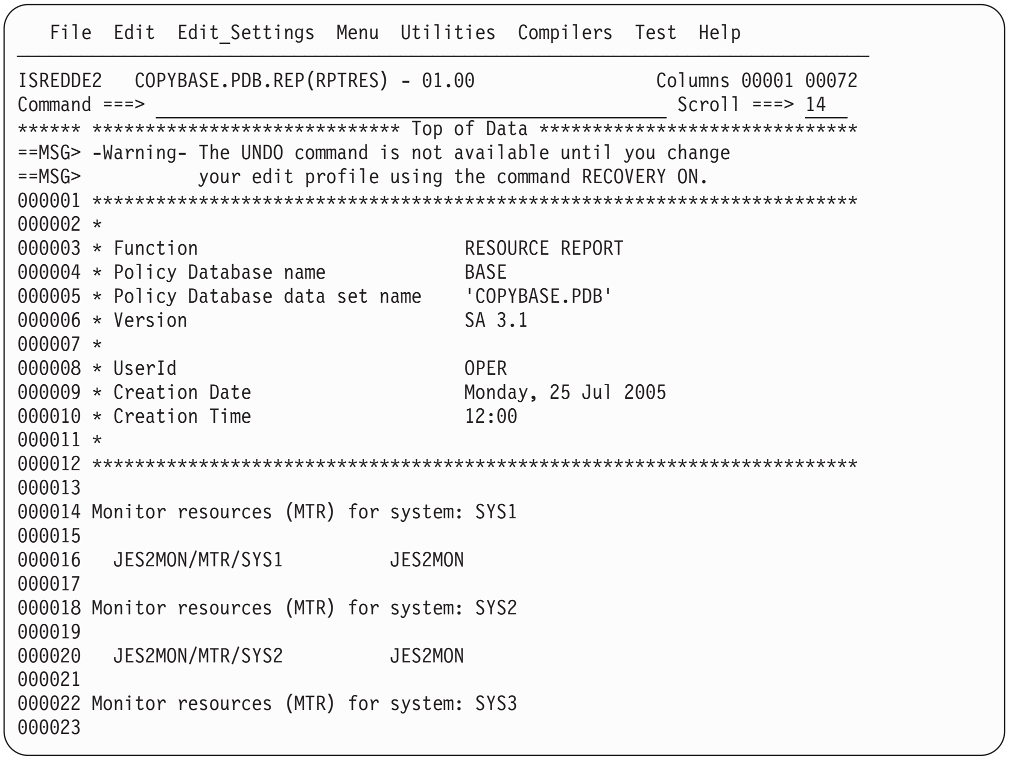
- all relationships that are not used during a build

It does not include the entry type Processor (PRO).

3 ViewUnlinked
Use this option to view the report of all unreferenced objects created by option *2 Unlinked*.

4 Resources
Use this option to create a list of all generated resources. In the Command Progress Display you will see in which member of the current policy database data set the generated list has been stored.

5 ViewResources
Use this option to view a list generated with option *4 Resources* for each existing system.



File Edit Edit_Settings Menu Utilities Compilers Test Help

```
ISREDDE2   COPYBASE.PDB.REP(RPTRES) - 01.00                 Columns 00001 00072
Command ===>                                                 Scroll ===> 14
****** ***************************** Top of Data ******************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG>           your edit profile using the command RECOVERY ON.
000001 ************************************************************************
000002 *
000003 * Function                        RESOURCE REPORT
000004 * Policy Database name            BASE
000005 * Policy Database data set name   'COPYBASE.PDB'
000006 * Version                         SA 3.1
000007 *
000008 * UserId                          OPER
000009 * Creation Date                   Monday, 25 Jul 2005
000010 * Creation Time                   12:00
000011 *
000012 ************************************************************************
000013
000014 Monitor resources (MTR) for system: SYS1
000015
000016   JES2MON/MTR/SYS1          JES2MON
000017
000018 Monitor resources (MTR) for system: SYS2
000019
000020   JES2MON/MTR/SYS2          JES2MON
000021
000022 Monitor resources (MTR) for system: SYS3
000023
```

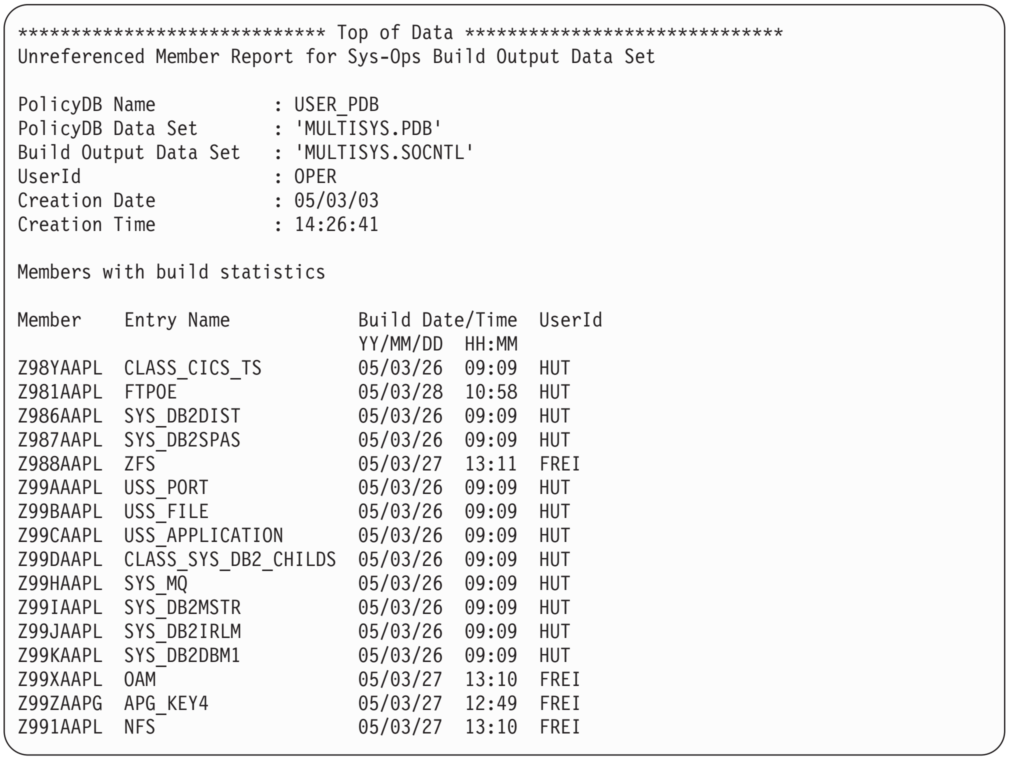
Figure 252. List of Generated Resources

6 MemberList
Use this option to create a report of all members in the system operations build output data set that are not referenced by the actual configuration anchored in the ACF map member AOFACFMP. This helps to identify members that can be deleted when you perform a data set cleanup. The report is divided in two sections:

1. The first section lists all unreferenced members that have been identified as valid configuration files, together with some additional information such as entry name, or build time.
2. The second section lists all other unreferenced members, where either no build statistics could be found, or which have not been identified as valid configuration files.

Report members created by the Customization Dialog itself such as build report member $BLDRPT are not included in the report. To run this report, a successful build is required. Otherwise the control file fragments may be damaged and incomplete, and the content of the report is unpredictable.

7 ViewMemberList
Use this option to read the report of all unreferenced members created by option 6.



```
***************************** Top of Data ******************************
Unreferenced Member Report for Sys-Ops Build Output Data Set

PolicyDB Name           : USER_PDB
PolicyDB Data Set       : 'MULTISYS.PDB'
Build Output Data Set   : 'MULTISYS.SOCNTL'
UserId                  : OPER
Creation Date           : 05/03/03
Creation Time           : 14:26:41

Members with build statistics

Member    Entry Name           Build Date/Time  UserId
                               YY/MM/DD  HH:MM
Z98YAAPL  CLASS_CICS_TS        05/03/26  09:09  HUT
Z981AAPL  FTPOE                05/03/28  10:58  HUT
Z986AAPL  SYS_DB2DIST          05/03/26  09:09  HUT
Z987AAPL  SYS_DB2SPAS          05/03/26  09:09  HUT
Z988AAPL  ZFS                  05/03/27  13:11  FREI
Z99AAAPL  USS_PORT             05/03/26  09:09  HUT
Z99BAAPL  USS_FILE             05/03/26  09:09  HUT
Z99CAAPL  USS_APPLICATION      05/03/26  09:09  HUT
Z99DAAPL  CLASS_SYS_DB2_CHILDS 05/03/26  09:09  HUT
Z99HAAPL  SYS_MQ               05/03/26  09:09  HUT
Z99IAAPL  SYS_DB2MSTR          05/03/26  09:09  HUT
Z99JAAPL  SYS_DB2IRLM          05/03/26  09:09  HUT
Z99KAAPL  SYS_DB2DBM1          05/03/26  09:09  HUT
Z99XAAPL  OAM                  05/03/27  13:10  FREI
Z99ZAAPG  APG_KEY4             05/03/27  12:49  FREI
Z991AAPL  NFS                  05/03/27  13:10  FREI
```

Figure 253. Unreferenced Member Report

8 StatusMsgs
Select this option to create a *Status Messages Report*. This includes Status Messages that are predefined by SA z/OS as well as user-defined Status Messages. The report shows the message ID, message description, and resource status indicated by this message and, if it is defined in the policy database, the entry name and the entry type that the message has been defined for. An example of the start of the Status Messages Report is shown in Figure 254 on page 335.

Note:
The status messages that are predefined by SA z/OS are not shown in the policy database report (option 1 on the SA z/OS Report Selection Menu shown in Figure 249 on page 329).

Pressing Enter returns you to the *Report Selection* panel from where you can view the report by selecting option 9.

9 ViewStatusMsgs
Select this option to view the Status Message Report that was created by using option 8. An example of the start of the Status Messages Report is shown in Figure 254 on page 335.

In the report, user-defined messages are listed with their associated entry name and entry type, whereas those messages listed without these columns are predefined SA z/OS messages.

```
/*********************************************************************/
/*                                                                   */
/*    Function : Status Messages Report                              */
/*                                                                   */
/*    PolicyDB                                                       */
/*      Name    : USER                                               */
/*      Data set: 'USER.MYPDB.PDB'                                   */
/*      Output  : 'USER.MYPDB.ACF'                                   */
/*                                                                   */
/*    User      : USER                                               */
/*    Date/Time : 2005/04/24 at 12:09                                */
/*                                                                   */
/*********************************************************************/


Message id/Description        | Entry Name        | Typ | Status
------------------------------+-------------------+-----+---------------
EMM2CLZ                       | EMMCLA            | APL | UP
AAA006I                       | EMM1              | APL | UP
XYZ444I                       | EMM1              | APL | UP
ATB010I                                                 | TERMINATING
BBMZA001I                                               | UP
BBMZA999I                                               | TERMINATED
BBOU0001I                                               | ACTIVE
           J2EE Control Region active
BBOU0002I                                               | TERMINATED
           WAS Control Regions ended normally
BBOU0003E                                               | BREAKING
           WAS Control Regions ended abnormally
BBOU0008I                                               | TERMINATING
           WAS DAEMON ended normally
BBOU0009E                                               | ABENDING
           WAS DAEMON ended abnormally
BBOU0016I                                               | UP
           DAEMON up
BBOU0020I                                               | UP
           SMS, Naming, IR, J2EE Control Region up
BBOU0021I                                               | UP
....
....
```

Figure 254. Example Status Messages Report (Excerpt)

A Symbols

This option allows you to create a report with information about all the system automation symbols (AOCCLONE*x*) in the policy database.

The report file is written to the report output data set (default name is the policy database data set name with .REP as the low level qualifier). The member name is RTPSYM.

The report includes:

- The Symbolname and Description, as defined in the Enterprise (ENT) SYMBOL DEFINITION policy
- For each Symbolname it shows the System name, Value and Specific Description as defined in the System (SYS) AUTOMATION SYMBOLS policy for each System entry where either the Value or Specific Description have been defined for that Symbolname

B ViewSymbols

Select this option to view the Symbols Report that was created by using option A. An example of the start of the Symbols Report is shown in

Figure 255.

```
************************************************************************
*
* Function                         SYMBOLS REPORT
* Policy Database name             BASE
* Policy Database data set name    COPYBASE.PDB
* Version                          SA 3.2
*
* UserId                           OPER
* Creation Date                    Monday, 11 Sep 2006
* Creation Time                    15:20
*
************************************************************************
*
-------- Symbolname ----------- |General Description
System              |Value      |Specific Description
*
-------- AOCCLONE  ------------ |MVS root suffix
DEMOMVS             |A          |
DEMOMVS2            |B          |
DEMOMVS3            |C          |
TESTMVS             |T          |
ZSERVERO            |T          |
*
-------- AOCCLONE1 ------------ |VTAM subarea
DEMOMVS             |16         |VTAM subarea
DEMOMVS2            |17         |
DEMOMVS3            |18         |
```

Figure 255. Example Symbols Report (Excerpt)

Chapter 8. Building and Distributing Configuration Files

After you have completed the SA z/OS customization process, the following data and control files can be built:

- Processor operations control file (needed for processor operations functions)
- System Operations configuration files (ACF, AMC, AT, and MPFLSTSA files), which are needed for automation
- NetView operator definitions

Note:
You must use the customization dialog to create the system operations configuration files. Do not edit any of the system operations configuration files manually.

How to Define Data Sets for Build Processing

To use the batch options of the build facility for automation control files, you must first specify the names of the data sets required for the batch processing. These names form the DD name concatenations to be included in the JCL of the batch processing job.

The data set names are saved in an ISPF table in the partitioned data set referenced by the AOFTABL DD name, for example, ING.CUSTOM.AOFTABL (allocated during installation of SA z/OS).

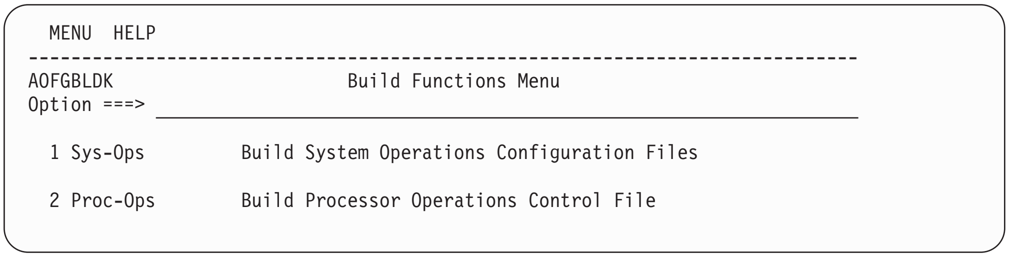
On the Customization Dialog Primary Menu panel of the customization dialog, select the *Settings* option. This displays the Customization Dialog Settings Menu (see Figure 277 on page 392).

On this panel, choose option `11, DSNAMES`. The Data Set Customization Details panel is displayed (see Figure 280 on page 396).

For more information, refer to “Defining Data Sets for Batch Processing” on page 396.

How to Build the Configuration Files

Exiting the customization dialog does not automatically build the configuration files. They are built by selecting the BUILD (B) function on either the OPTIONS menu of the Entry Type Selection panel or the COMMANDS menu of the Policy Database Selection panel. The panel in Figure 256 on page 338 is displayed.



```
  MENU  HELP
------------------------------------------------------------------------------
AOFGBLDK                      Build Functions Menu
Option ===> __________________________________________________________________

  1 Sys-Ops         Build System Operations Configuration Files

  2 Proc-Ops        Build Processor Operations Control File
```

Figure 256. Build Functions Menu Panel

The Build Functions Menu offers the following functions to create control files:

1. **Build the System Operations Configuration Files**

   This causes enterprise automation data for SA z/OS system operations to be built. This data consists of:

   - The automation control file (ACF), which is built into a set of file fragments and a master include file
   - The automation manager configuration file (AMC file)
   - The NetView automation table (AT)
   - The MPFLSTSA member

   For more details see "Building the System Operations Configuration Files" on page 339.

2. **Build the Processor Operations Control File**

   This causes the information that is required by processor operations to be generated in a processor operations control file. It also provides an option to specify the name of the message log for the processor operations control file generation process.

   The processor operations control file should be rebuilt after any changes to the definitions of enterprise, groups, subgroups, systems, processors, communications paths, or communications tasks.

   The processor operations control file should be built prior to building the system operations configuration files because the name of the processor operations file to be used for automation is placed in the configuration file fragment for enterprise data.

   For more details see "Building the Processor Control File" on page 347.

Figure 257 shows the files that are built.

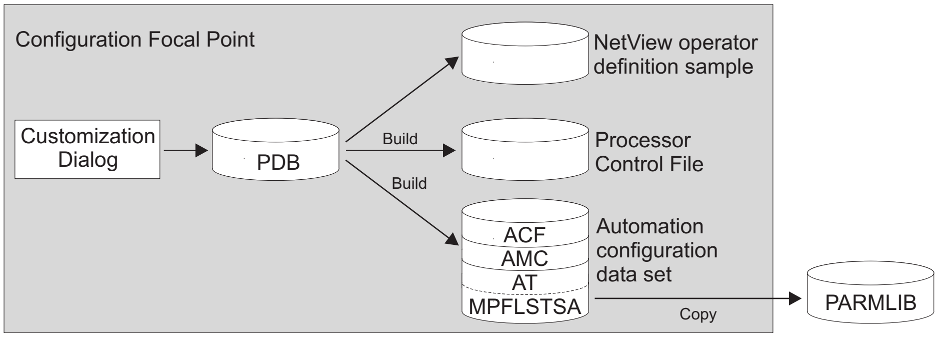

Configuration Focal Point
Customization Dialog
PDB
Build
Build
NetView operator definition sample
Processor Control File
ACF
AMC
AT
MPFLSTSA
Automation configuration data set
Copy
PARMLIB

Figure 257. Configuration File Build

The enterprise wide policy database for your sysplex and the standalone system is created and maintained with the customization dialog on any suitable system. In Figure 257 on page 338, it is called the *configuration focal point*.

The configuration build is also performed with the customization dialog and the configuration files are distributed from there to the locations where needed.

Using the customization dialog's BUILD function, you produce the following output:

- The Processor Operations Control File
- The System Operations Configuration Files

Building the System Operations Configuration Files

To build a new enterprise automation configuration, that is, a set of new system operations configuration files (automation control file, automation manager configuration file, NetView automation table, and MPFLSTSA member), you can invoke the BUILD process in the following ways, depending on which panel you start from:

- From the **Policy Database Selection** panel, enter the action code B in the **Action** column of the policy database that you want to invoke the BUILD process for. This takes you to the Build Functions Menu (see Figure 256 on page 338).
- Use option 1 on the Build Functions Menu (Figure 256 on page 338). This will bring you to the Build Parameters panel shown in Figure 258 on page 340.
- From the **Entry Type Selection** panel, you can invoke the build functions as follows:
  - Choose the BUILD (B) command from the **Command** menu. This will bring you to the Build Functions Menu.
  - If you want to process a certain entry type, enter the BUILDF (F) command at the command line and specify the identifier for the entry type that you want to build. For example, to build all defined system objects, enter: `BUILDF SYS` or `F SYS`. You then proceed to the Build Parameters panel (see Figure 258 on page 340) with the entry fields filled in according to the parameters that you entered for the BUILDF command.

  Note that the build function builds all systems irrespective of whether they have been modified or not.
- From the **Entry Name Selection** panel, you can enter the BUILDF (or F) command in the **Action** column against the object you want to build. For example, to build the JES2 Subsystem enter F against JES2 in the Entry Name Selection panel for Application policy objects.

  Optionally, you can enter the command at the command line, for example:

  ```
  F JES2
  ```

  You then proceed to the Build Parameters panel (see Figure 258 on page 340) with the entry fields filled in according to parameters that you entered for the BUILDF command.

A *Command Progress Display* panel shows the progress of the build. If errors are encountered during the build process, the Command Progress Display will display a message. The automation configuration files are placed in the data sets that you defined in the *Output Data Set* field.

Every new build generates an automation configuration map AOFACFMP that contains

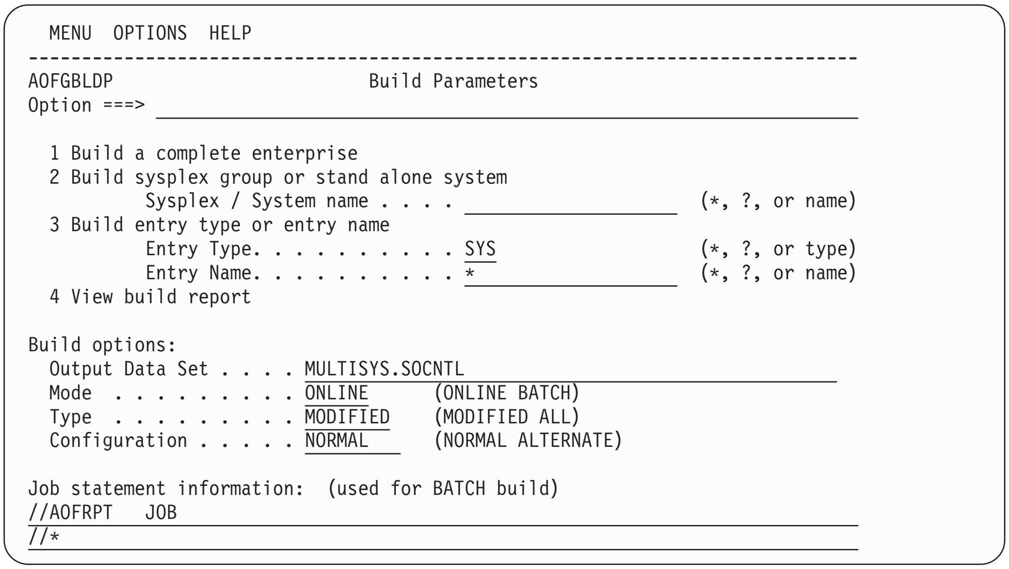
- For each system one ACF*xnnn* system include member that is used by the automation agents
- For each sysplex one HSA*xnnn* sysplex include member that describes the automation manager configuration

The ACF, AMC, AT, and MPF build will always be invoked together. This is because the configuration data always needs to be synchronized so that the automation agents and the automation manager operate on the same data.

Each automation control file fragment contains information about a specific resource that is defined through the customization dialogs. Each z/OS system in the enterprise has its own master include file. The master include file references the automation members that are appropriate for that system. The name of the master include file for each system is created by SA z/OS. It starts with prefix ACF followed by four alphanumeric characters.

System Operations Configuration Files Build Options

The Build Parameters panel allows you to specify the information required to build the system operations configuration files that are required for automation. You can build these system operations configuration files for standalone systems, individual sysplexes or your complete enterprise, using the options shown in Figure 258.



```
  MENU  OPTIONS  HELP
------------------------------------------------------------------------------
AOFGBLDP                        Build Parameters
Option ===>

  1 Build a complete enterprise
  2 Build sysplex group or stand alone system
         Sysplex / System name . . . .                      (*, ?, or name)
  3 Build entry type or entry name
         Entry Type. . . . . . . . . . SYS                  (*, ?, or type)
         Entry Name. . . . . . . . . . *                    (*, ?, or name)
  4 View build report

Build options:
  Output Data Set . . . . MULTISYS.SOCNTL
  Mode  . . . . . . . . . ONLINE      (ONLINE BATCH)
  Type  . . . . . . . . . MODIFIED    (MODIFIED ALL)
  Configuration . . . . . NORMAL      (NORMAL ALTERNATE)

Job statement information:  (used for BATCH build)
//AOFRPT   JOB
//*
```

Figure 258. Build Parameters Panel

You can enter one of the following build processes:

1 Build a complete enterprise
Builds all System Operations automation configuration files for a complete enterprise.

Note:
This option must be successfully completed at least once before you can use options 2 or 3.

2 Build sysplex group or stand alone system

Allows the selection of a single sysplex group or a standalone system.

This build option performs a complete build of all objects linked to the selected sysplex or standalone system. For a large policy database, this options helps to reduce build time for small changes, while creating a complete set of configuration files that are ready for automation.

3 Build entry type or entry name

Allows you to specify an entry type and name to be rebuilt. Only the ACF fragments for that entry will be updated.

If for example, a timer object is modified and linked to another system, a build of this object with option 3 would only build that fragment and update the timestamp in the system include member, but not build the link changes for the system.

The same is true if an object is renamed. A build with option 3 of the new name would not add this new object to the system include member. Instead a build with option 2 is required.

For entry type System, this has the consequence that a selective build is rejected if there is also a link change pending.

However, if this option is used with entries of type Events, Triggers, Service Periods, Application, ApplicationGroup, System, or Group, a check is made as to whether data that affect the automation manager has been changed.

For certain entry types (for example, Service Periods, Events, Triggers) this option cannot be used because they do not have data included that needs to be built into an ACF fragment. In this case, building only the ACF fragment for the requested entry would mean that your previously generated AMC files would no longer be compatible with your current ACF files. To avoid this, a complete build (using build type MOD) will automatically be invoked. Before the build actually starts a confirmation panel is displayed explaining this situation, and allowing you to defer the build.

You will see the same behavior with automation tables if this option is used with entries of type APL or MVC. A check is made as to whether data has been changed that affect the NetView automation table. If so, a confirmation panel is displayed allowing you to defer or to start a complete build (using build option MOD).

If you enter a ’?’ in this field, SA z/OS displays a list of either entry types or a list of available entry names for the selected entry type, or both.

4 View build report

Allows you to view the build report available on the build output data set. It invokes the ISPF VIEW function. If there is no build report, an empty screen is displayed.

Notes:

1. If a policy database is new, or the build output data set is empty, a full build is required to produce a complete set of system operations configuration files. This means a build with option *1* and `Type = ALL` needs to be issued. For all subsequent builds, option *2* or *3* and `Type = MODIFIED` can be used.
2. While build option *2* only updates a single system or sysplex, deleted standalone systems and sysplexes still remain in the configuration files. For example, a deleted sysplex or standalone system is not removed from the ACF map AOFACFMP. So, you may consider issuing a build with option *1* whenever a large number of parts has been deleted, to clean up unnecessary entries from the configuration files.
3. Fragments are not built for all entry types (for example, not for SVP or TRG) or all entries in an entry type (for example, not for PRO entries of type PSM).

In the **Build options** section, you can enter values for the following fields:

Output Data Set

Use this field to specify where you want to store the build output. The data set must be an existing partitioned data set.

If the data set is not empty, but already contains a control file, the policy database name and the configuration name within the data set are compared with those of the requested build. If they do not match, a message is issued and the build is not started.

Mode

Use this field to specify how to process the BUILD request:

ONLINE

Indicates that the BUILD should run in the foreground.

BATCH

Indicates that the BUILD should run in the background.

The default value is ONLINE (if this field is set to blank).

Type

Use this field to specify which entries are to be processed.

MODIFIED

Indicates that only those entries that have been changed since the last BUILD request should be processed.

With this option, the build distinguishes between:
- Data changes in the ACF
- Link changes in the ACF
- Data changes in the AMC file
- Link changes in the AMC file

Depending on the type of change, the affected objects are identified, and only the necessary parts of the configuration file are rebuilt.

For example:

- If a new *timer* is linked to one system, only the system include member for that system is rebuilt. For all other system include members only the header timestamps are updated, and the build of the AMC file, AT, and MPFLSTSA member is skipped, because timer data does not go to the automation manager configuration file, AT, or MPFLSTSA member.

- If a new *application* is linked to a Sysplex Application Group, the build determines which systems are linked to that sysplex, and the include members for those systems are rebuilt automatically, together with the AMC, AT, and MPFLSTSA member for that sysplex.
- If the AT Scope is changed, only the AT and MPFLSTSA member are built.

ALL Indicates that all entries are to be processed unconditionally, even if there has been no change to those entries since the last BUILD request.

If a build type ″ALL″ is selected together with build option 2, only the configuration files belonging to the selected sysplex or standalone system are rebuilt.

The default value is MODIFIED (if this field is set to blank).

The very first build after installing a new release must be a build of type ALL in order to get a complete set of configuration files for the automation manager and automation agent structure.

Configuration

This field allows you to select between alternative configurations:

NORMAL

indicates that only those entries linked to the systems of the NORMAL configuration are processed by the current build.

ALTERNATE

indicates that only those entries linked to the systems of the ALTERNATE configuration are processed by the current build. These systems must have been marked with a ″+″ in their MVS system name.

NORMAL and ALTERNATE configurations must be built into different data sets. The default value is NORMAL (if this field is set to blanks).

In the **Job statement information** section you must enter a valid JOB statement for the batch job, if you specified that the BUILD request should run in BATCH mode.

The information that you specify here is saved in your ISPF profile data set and is used as the initial values the next time you issue the BUILD command.

BUILD Error Messages

The messages in the popup panel showing the command progress indicate error and warning situations. In most cases, it should not be necessary to check the build output member for unusual entries or error indications. Furthermore, the message log can be viewed from the panel where the build has been issued.

There are build errors that may stop the build, for example, resources in a loop, and other messages that are tolerated, for example, duplicate MVS jobnames linked to the same system. Terminating build errors are shown with the label *ERROR* in the message prefix section, while tolerated errors are shown with the label *WARNING* in the message prefix section.

When an error is shown, no usable set of configuration files has been created. It is necessary to correct the cause of the problem and run the build again.

When a warning is shown, the configuration files can be used for automation. However it is recommended that you fix the problem and run the build again.

A possible cause for a warning during ACF fragment build may be that there are policy database objects with missing or conflicting definitions, that are not critical for automation, for example, an application default entry (type ADF) without any definitions. In this case, the build function continues, but issues a warning message. This message is displayed whenever a full enterprise build is issued, and does not require that the application be linked to a system.

There is a summary message at the end of the AT build report, and specific messages that show, for example, the object name somewhere within the build progress messages. The specific message reads:

```
.......WARNING. System MIGR_TEST includes the following entries with
..........cont. an unsatisfactory build status in main member ACFZ995:
............... -- SYSTEM_DEFAULTS  (NUL)
```

The summary message at the bottom of the build message log reads:

```
.......WARNING. Missing or conflicting definitions found for entry
..........cont. types: SDF. The entry names are listed in the build
..........cont. report.
```

Possible causes for a warning during system main include member build are:

- Conflicting type definitions for an application instance that is linked to an application class
- Applications with duplicate MVS jobnames linked to the same system. The processing is the same as for duplicate subsystems.
- If missing or conflicting definitions are detected, which are not critical for automation, a warning message is shown and the entry name is listed together with the status *NUL*. There are only a few situations where this can occur, for example, an application default entry (type ADF) without any definitions.
- If an object is linked and has been previously built, however it does not exist in the build output data set, a warning message is shown and the entry name is listed together with the status *DEL*. The build function continues, but the object is not included in the system operations control files.

Note that such a warning message may cause a rebuild of the objects in error and the linked systems and sysplexes. So, a build of type MOD may rebuild more systems and sysplexes than you would expect due to the actual changes. Possible causes for a warning during automation manager configuration file build are:

- The AMC logic member could not be found. The build function searches in the sample PDB data set for the current logic member. If this data set is not allocated, or contains no logic member, it could be added to the Configuration Files by the user.
- A trigger is linked to the current sysplex, but no conditions have been specified. The trigger will not be added to the automation manager configuration file.
- One or more application groups are linked to the current sysplex without an automation name being specified. The build function continues, but no resources will be created for this application group(s).
- One or more supporting application resources are not linked to the system. The relationships are not added to the automation manager configuration file.

Possible causes for a warning during AT and MPFLSTSA build are:

- Conflicting status message definitions for the same message ID but on separate policy database entries
- AT entry override definitions for the same message ID for one policy database entry but a status message definition for different policy database entries
- The AT scope being set to SYSPLEX but the changes that affect the AT being made for systems that are not linked to any sysplex

It is possible to tailor the level of detail of the messages for the trigger, application group, and AT warning. When invoking option *0 Settings*, there is a selection *3 MESSAGE* that allows the message level to be set. With level 1 (which is the default), in the above situation just one summary message is displayed per manager configuration, whilst level 2 will display all details of the definitions causing the problem.

Furthermore, that panel offers an option for message logging. By default, this value is *ON*, which means that all important dialog messages, including messages shown in a command progress popup window, are written into the ISPF LOG. The value *OFF* suppresses this, and it may be useful to switch this message logging off, if a build is run with message level 2.

Independent from the message logging switch, error messages are always written to the ISPF LOG. You cannot turn this off.

If the user profile AOFPROF for batch build is not allocated, the default message level is 3 to write all possible messages to the batch report and the ISPF LOG.

Distributing the System Operations Configuration Files

The SA z/OS automation configuration consists of four components:

- The automation manager's automation manager configuration file (AMC)
- The automation agent's automation control files (ACF)
- The NetView automation table (AT)
- The MPFLSTSA member

These files are also referred to as *system operations configuration files* in the customization dialog.

The partitioned data set generated by the BUILD contains these four components. SA z/OS runs on a sysplex basis, that is, the same ACF, AMC, and AT files must be available to all of the automation agents and automation managers in the sysplex.

The automation agent automatically finds the automation configuration data set and dynamically allocates it. It is highly recommended that you do not allocate the data set within DSIPARM. (For further information, refer to the section that describes using SA z/OS partitioned data sets in *IBM Tivoli System Automation for z/OS Planning and Installation*.) The automation configuration data set must either be shared or have exactly the same name on all systems within a sysplex. It can be shared within an enterprise.

You then have the choice to:

- Copy the members into the initial configuration data set that the automation manager was started with. Then issue an INGAMS REFRESH command with an * as the data set name.

- Issue an INGAMS REFRESH command with the name of the new partitioned data set. This will cause the automation manager and the automation agent to read from the new partitioned data set.

With SA z/OS the Config Dataset is dynamically allocated from the Automation Manager and from the Automation Agent.

This enables a sort of 'change management' by using multiple data sets or GDG versions. A simple and controlled way to switch configurations is with the INGAMS command. Using dynamic allocation and multiple data sets (or GDG versions) is recommended, because it provides maximum flexibility, availability and reliability for your automation. A 'token mismatch' normally cannot occur.

An automation agent needs to run on every system that is automated by SA z/OS and at least one automation manager needs to run per sysplex and one per standalone system. Automation managers can manage systems sysplex-wide. Figure 259 shows a sample automation configuration and presents the build process.

The MPFLSTSA member must be copied to a z/OS parmlib data set.

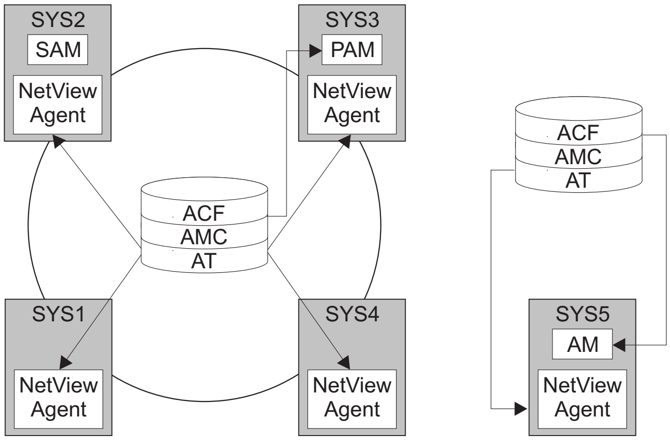


Figure 259. SA z/OS Automation Configuration

Figure 259 shows a sample enterprise consisting of one sysplex Sysplex1 and one standalone system SYS5. The sysplex consists of four systems SYS1, SYS2, SYS3, SYS4 and has a primary and a secondary automation manager (PAM and SAM). The sysplex shares a copy of the system operations configuration files and the standalone system has its own copy of the same files.

The sysplex in this example has one primary and optionally one secondary automation manager and one automation agent on each system. The standalone system must have one automation manager (AM) and one automation agent.

You could share one common automation configuration file in your enterprise, or you can copy the automation configuration file to where it is needed as shown in our scenario. Both automation managers feed the enterprise-wide RODM data base dynamically with information that is available from both AMCs and from information that becomes available during automation processing (like status changes). This feeding is done automatically by the SA z/OS topology manager after you have started it. The generated data is used by the NMC workstation to display predefined graphical views.

The automation manager's AMC file and the automation agent's ACF files must match. Therefore the automation manager and an automation agent exchange a token to verify that the automation configuration matches. You can share one common data set containing all automation manager configuration files and all automation control files within a sysplex.

Note:

It is mandatory to create and maintain policy databases with the customization dialog.

The automation configuration files must be distributed to each system in the SA z/OS enterprise where SA z/OS system operations functions are to run. In general, this is all of the z/OS systems in the SA z/OS enterprise. In a sysplex, the data set name of the automation configuration file must be equal on all systems in the sysplex.

However, it is highly recommended that you use a shared data set in a sysplex, and it is preferable that you use a shared data set for the whole enterprise.

Generating a Job for Batch Build

You can run a job for batch build that performs the same processing as the BUILD function of the customization dialog. The SINGSAMP sample library contains a sample job for this task in member INGEBBLD. Before submitting this job, you need to edit the jobcard appropriately as described in the sample itself.

The build can only run if the output data set exists and is allocated. If the data sets are not SMS-managed, allocating the data set within the same job step does not work because the cataloging is done by z/OS at end of the job (or job step) . Therefore the allocation and cataloging of the build output data set must be done in a separate job (or job step) prior to the build job (or job step).

The batch build produces the following return codes:

0 Successful

4 Build complete, however, some warning messages are issued. Also issued if user exit INGEX02 ended with a RC > 0.

8 User exit INGEX01 ended with RC > 0. Build processing not started. No configuration files built.

16 Build not started. The policy database is from a previous version of SA z/OS. The policy database must be converted first before a batch build will be successful.

20 Severe error; build terminated without usable configuration.

SA z/OS provides user exits at invocation and termination of the build function. For more details, see "User Exits for BUILD Processing" in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

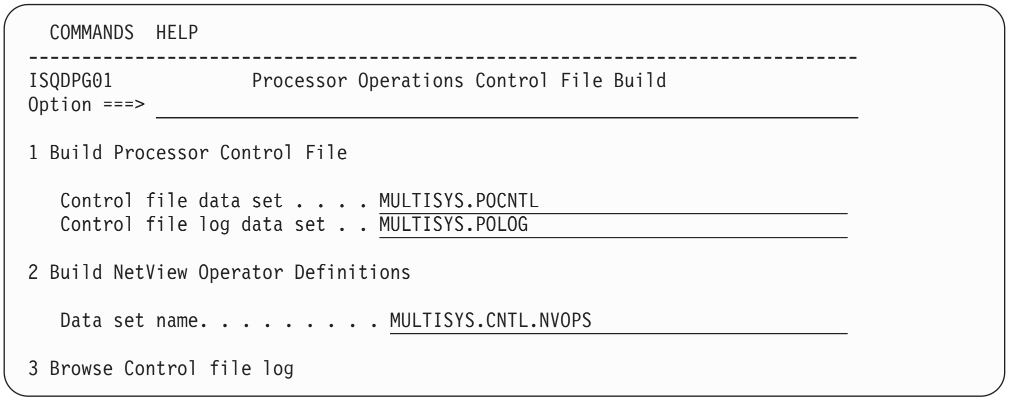
Building the Processor Control File

To create the processor control file, you can use option 2 on the Build Functions Menu.

Building the Processor Control File

When you request a build of the processor control file, it is built from information on the enterprise, groups, subgroups, systems, and processors, already entered in the policy database. This means that any time you change definitions of any of these, the processor control file needs to be rebuilt.

Use the panel shown in Figure 260 to identify the fully qualified names of both the SA z/OS processor control file and the control file message log.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
ISQDPG01            Processor Operations Control File Build
Option ===> __________________________________________________________________

1 Build Processor Control File

   Control file data set . . . . MULTISYS.POCNTL______________________________
   Control file log data set . . MULTISYS.POLOG_______________________________

2 Build NetView Operator Definitions

   Data set name. . . . . . . . . MULTISYS.CNTL.NVOPS__________________________

3 Browse Control file log
```

Figure 260. Processor Operations Control File Build Panel

For SA z/OS processor operations, the processor control file and control file log must be in a sequential data set.

This panel has the following sections:

1 Build Processor Control File
Builds the ProcOps Control File and writes all build messages in a separate log file. The following needs to be specified:

Control file
Identifies the data set containing the ProcOps Control File. This data set must be preallocated. You can use sample job INGEDLGA to allocate it (see *IBM Tivoli System Automation for z/OS Planning and Installation*).

Control file log
Identifies the data set containing information about a successful build or errors in the configuration definition. This data set must be preallocated. You can use sample job INGEDLGA to allocate it (see *IBM Tivoli System Automation for z/OS Planning and Installation*).

2 Build NetView Operator Definitions
Builds a sample NetView Operator Definitions file. The number of target control tasks and message monitor tasks is calculated as the number of current processors that are defined with a connection protocol other than INTERNAL.

The file can be used to update the AOFOPFPO sample that is included in the NetView DSIOPF member.

Data set name
Identifies the data set containing the NetView Operator Definitions sample file.

3 Browse Control file log

Allows you to view the ProcOps control file log containing all messages sent from the control file generation process. The log is accessed in browse mode.

Note:

For the resulting NetView operator definition file, each message monitor task and target control task is a NetView autotask. These autotasks must be defined to the NetView program to enable SA z/OS processor operations.

To retain any operator and autotask definitions that you already use, make sure you generate the SA z/OS definitions into a new data set or member. The generate function erases any statements in the chosen data set. Integrate the contents of the new NetView operator definition file into the DSIOPF member of the NetView DSIPARM data set. The statements can be anywhere in the NetView DSIOPF member.

The customization dialog generates automation operator definitions with an operator profile of ISQPROF. ISQPROF is provided in the SINGNPRF data set and should be added as a member of a data set in the DSIPRF path. If your data center has changed the ISQPROF profile name, you must change the profile name in each of the automation operator definition statements.

Usage Notes

The SA z/OS processor control file contains information specific to your SA z/OS configuration and is used to start SA z/OS processor operations. Before you generate the processor control file, you must allocate and catalog the data sets for it and its log. The INGEDLGA job does this for you early in the installation process. If you decide to use more than one control file before you generate each processor control file, you must allocate and catalog the data sets for them and their logs.

If the control file you specified already contains data, a confirmation panel is displayed to allow you to confirm whether you want the existing control file to be overwritten.

If the data set name of either the control file or control file log has been changed (using PF3), the values are stored into the enterprise policy database. To avoid this, exit the panel with `CANCEL`.

The processor operations control file should be built prior to building the automation control file because the name of the processor control file to be used for automation is placed in the ACF.

The enterprise member of the automation control file should be rebuilt after any change to the name of the processor control file. This name is used in the automated startup of the processor operations application.

After you have used the generation function, check the Control File Log panel by selecting the *Browse Control file log* option on the *Processor Operations Control File Build* panel. This is option 3 in Figure 260 on page 348. To determine the correct response for any processor control file log messages, see *IBM Tivoli System Automation for z/OS Messages and Codes*.

Once you complete the actions recommended for all of the messages, generate the processor control file again.

The control file log receives messages from the control file generation process. These messages indicate why an attempt to generate a control file produced an incomplete control file or was unsuccessful. When a control file is generated successfully, a message to that effect is displayed, and no messages are written in the control file log.

To successfully generate an SA z/OS control file, you must have completed the required fields in all the configuration panels. If you try to generate a processor operations control file for incomplete definitions, a message is displayed that the control file is incomplete or was not generated. When this occurs, examine the control file log to determine what needs to be corrected.

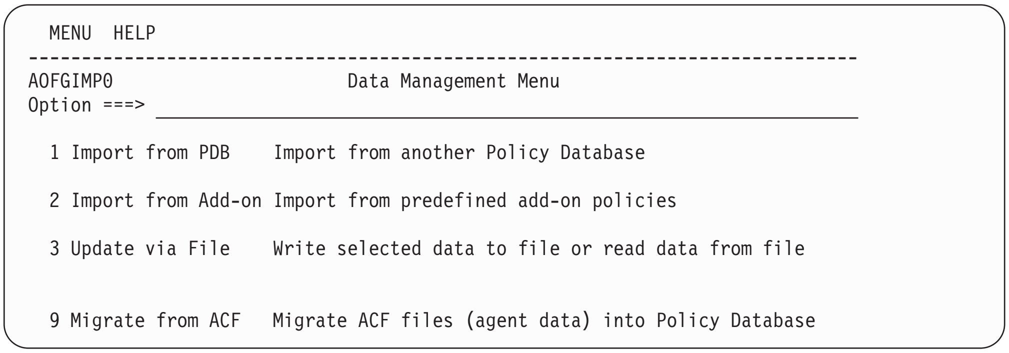
Chapter 9. Data Management

This chapter provides information about SA z/OS functions for:

- Importing selected entries from another policy database,
- Importing sample add-on policies,
- Making bulk policy updates via flat files,
- Migrating agent data from automation control files into a policy database

Start Panel for Data Management

Option 5 (Data Management) on the Customization Dialog Primary Menu takes you to the Data Management Menu panel, shown in Figure 261.



```
  MENU  HELP
------------------------------------------------------------------------------
AOFGIMP0                      Data Management Menu
Option ===> ___________________________________________________________________

  1 Import from PDB    Import from another Policy Database

  2 Import from Add-on Import from predefined add-on policies

  3 Update via File    Write selected data to file or read data from file


  9 Migrate from ACF   Migrate ACF files (agent data) into Policy Database
```

Figure 261. Data Management Menu Panel

Your options on this panel are:

1 Import from PDB
This option enables you to select entries in another policy database and add them to the current policy database. See “Importing Policy Database Data” on page 352 for details.

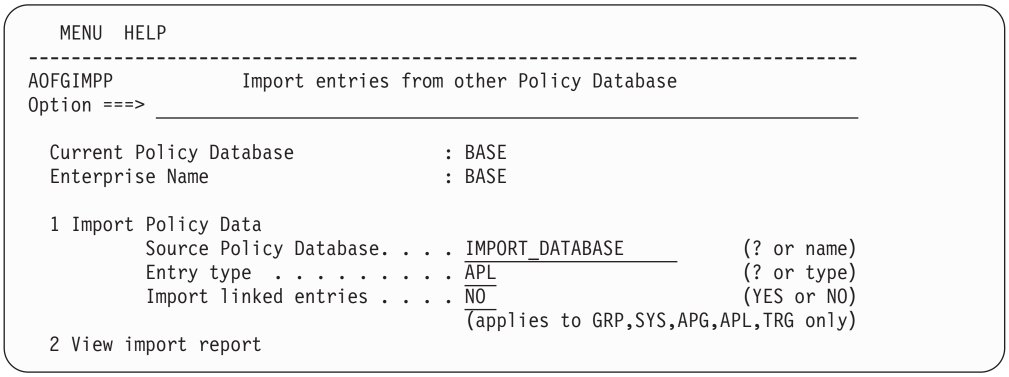
2 Import from Add-on
This option enables you to select sample add-on policies and add all or part of their entries to the current policy database. See “Importing Sample Add-On Policies” on page 354 for details.

3 Update via File
This option enables you to export the policy database data of a selected entry type to a flat file. After you have modified this data or added new entries, the flat file can be read in again. All the modified or newly added entries will be updated in the policy database or added to it without any further panel interaction. See “Making Updates with Text Files” on page 356 for details.

9 Migrate from ACF
This option enables you to migrate an Automation Control File (Agent Data) or Automation Control File fragments (ACF fragments), that is automation agent data *only*, into an existing policy database. See “Migrating the Automation Control File” on page 364 for details. Note that this migration function only accepts fragments that are built from SA z/OS 3.1 or earlier.

Importing Policy Database Data

Selecting option 1 (Import from PDB) on the Data Management Menu panel leads to the panel shown in Figure 262.



```
   MENU  HELP
 ------------------------------------------------------------------------------
 AOFGIMPP            Import entries from other Policy Database
 Option ===> ___________________________________________________________________

   Current Policy Database              : BASE
   Enterprise Name                      : BASE

   1 Import Policy Data
            Source Policy Database. . . . IMPORT_DATABASE          (? or name)
            Entry type  . . . . . . . . . APL                      (? or type)
            Import linked entries . . . . NO                       (YES or NO)
                                          (applies to GRP,SYS,APG,APL,TRG only)
   2 View import report
```

Figure 262. Policy Data Base Import Panel

This panel enables you to select entries in another policy database and add them to your current one, and view the import report for the target policy database. It has the following options:

1 Import Policy Data

This option allows you to define the following:

- An existing policy database that data will be imported from.

  Entering a question mark (?) in this field and pressing Enter displays the Policy Database Selection panel where you can select one of the existing policy databases.
- The particular entry type that selected entries will be imported for.

  Entering a question mark (?) in this field and pressing Enter displays the Entry Type Selection panel where you can select one of the existing entry types.
- Whether you want to import entries that are linked to the entry type that you selected in the **Entry type** field.

  The following rules determine which linked entries are also imported, depending on the entry type that is imported:

  GRP Linked entry types SBG or SYS are also imported. For sysplex groups, any linked APG entry type of type SYSPLEX is imported.

  The links between resources where the group is involved are imported.

  SYS All entry types that are linked to the selected systems are imported along with the resources that belong to those systems and the links between the resources in the systems.

  APG Linked entry types APL, SVP or TRG are also imported.

  APL If the application has an object type of CLASS, its instances and their service periods, triggers, CICS state/action and IMS state/action definitions are imported.

  If the application has an object type of INSTANCE, the class application that it is linked to and its service periods, triggers, CICS state/action and IMS state/action definitions are imported.

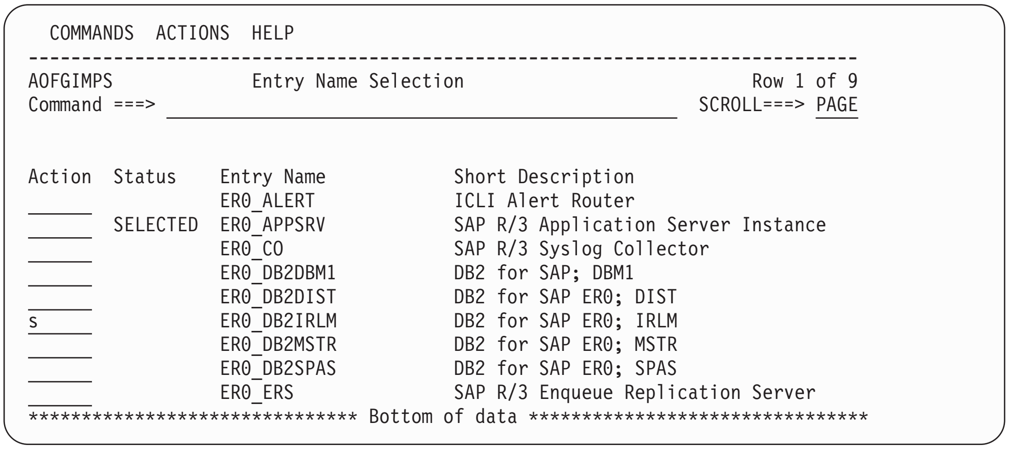
TRG The entries of entry type EVT that are selected in its startup or shutdown conditions are imported along with it.

> **Note:**
> Importing entry type ENT (enterprise policies) is not supported.

2 View import report
This allows you to view the import report that is available in the target policy database. It invokes the ISPF VIEW function. If there is no import report, an empty screen is displayed.

The information that you specify here is saved in your ISPF profile data set and used as the initial values the next time you issue the import function.

Selecting **1 Import Policy Data** on this panel takes you to a list of existing entries for the specified type (see Figure 263 for an example).



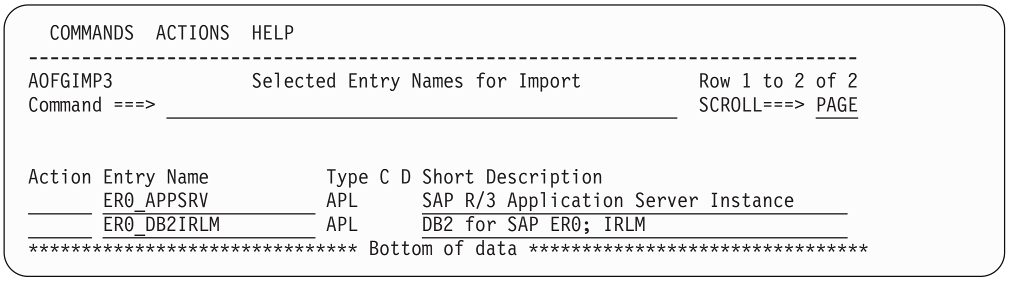
```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGIMPS             Entry Name Selection                           Row 1 of 9
Command ===> ______________________________________________ SCROLL===> PAGE


Action  Status    Entry Name           Short Description
______            ER0_ALERT            ICLI Alert Router
______  SELECTED  ER0_APPSRV           SAP R/3 Application Server Instance
______            ER0_CO               SAP R/3 Syslog Collector
______            ER0_DB2DBM1          DB2 for SAP; DBM1
______            ER0_DB2DIST          DB2 for SAP ER0; DIST
s_____            ER0_DB2IRLM          DB2 for SAP ER0; IRLM
______            ER0_DB2MSTR          DB2 for SAP ER0; MSTR
______            ER0_DB2SPAS          DB2 for SAP ER0; SPAS
______            ER0_ERS              SAP R/3 Enqueue Replication Server
****************************** Bottom of data ********************************
```

Figure 263. Entry Name Selection Panel for Import Policy Data

Here you can select entries that you want to:

- Copy to your current policy database
- Get a *modified* copy of for your current policy database

Making selections and pressing Enter takes you to the panel shown in Figure 264, which displays those entries that you selected.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGIMP3             Selected Entry Names for Import          Row 1 to 2 of 2
Command ===> ______________________________________________ SCROLL===> PAGE


Action Entry Name          Type C D Short Description
______ ER0_APPSRV          APL      SAP R/3 Application Server Instance
______ ER0_DB2IRLM         APL      DB2 for SAP ER0; IRLM
****************************** Bottom of data ********************************
```

Figure 264. Selected Entry Names For Import Panel

This panel allows you to modify the selected entries before you import them. You can:

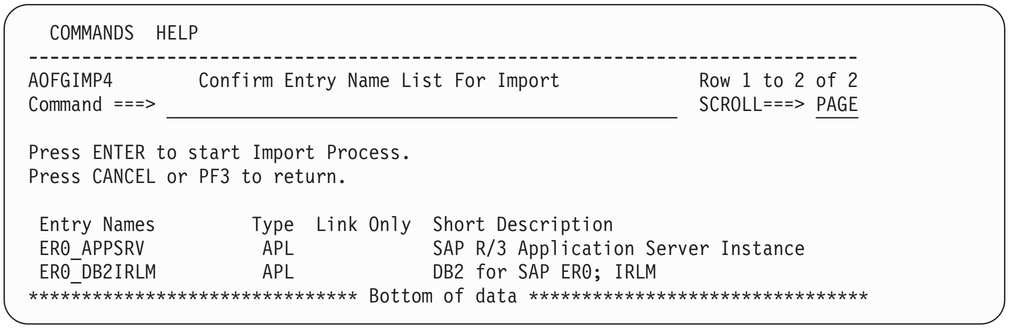
- Press Enter if the list is okay.

- Rename entries in the list by:
  - Overtyping their names
  - Requesting RENAME(R) in the Action field

  If an entry of same name and type exists already in the target policy database, it will be marked as duplicate (D is set to “Y”).
- Overtype the entry description field (effects non-duplicate entries only).
- Remove entries from the list by:
  - Blanking their names
  - Requesting REMOVE(M) in the Action field
- Locate an entry by entering LOCATE (L) on the Command line.
- Return to the previous panel by pressing END or entering the CANCEL command.

Note: If the entry is a class, column C is set to “*”.

When you are done with modifications and have pressed Enter, a panel is displayed where you are asked to confirm them, as shown in Figure 265.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGIMP4       Confirm Entry Name List For Import            Row 1 to 2 of 2
Command ===> ________________________________________________  SCROLL===> PAGE

Press ENTER to start Import Process.
Press CANCEL or PF3 to return.

 Entry Names         Type  Link Only  Short Description
 ER0_APPSRV           APL             SAP R/3 Application Server Instance
 ER0_DB2IRLM          APL             DB2 for SAP ER0; IRLM
******************************* Bottom of data ********************************
```

Figure 265. Confirm Entry Name List For Import Panel

Pressing Enter will start the import processing, whereas CANCEL or PF3 will take you back to the Policy Data Base Import panel shown in Figure 262 on page 352.

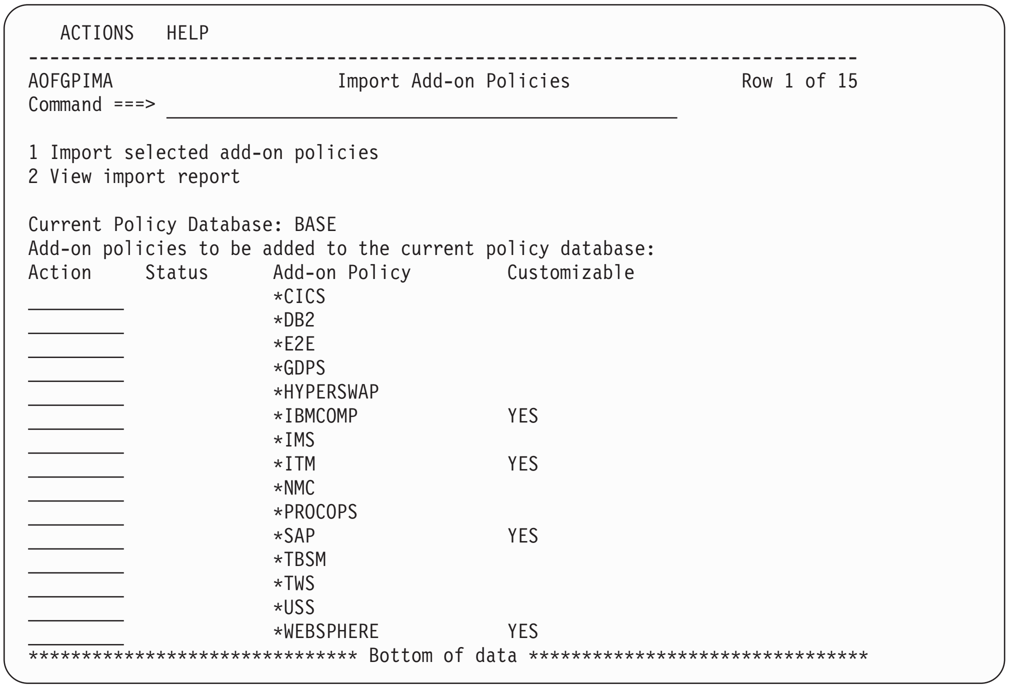
Importing Sample Add-On Policies

SA z/OS comes with a number of sample add-on policies that you can import into an existing policy database to speed up database development. Importing add-on policies into a policy database means adding a copy of the sample to that policy database. The add-on policies themselves are not modified.

To import sample add-on policies:

1. Specify the *target* policy database that you want to import the sample *into* in the **Current Policy Data Base** field of the Customization Dialog Primary Menu and select option 5 *(Data Management)*.
2. On the Data Management Menu panel, select option 2 *(Import from Add-on)* .

   This takes you to the Import Add-On Policies panel, shown in Figure 266 on page 355, which displays a list of the add-on sample policies that SA z/OS provides.

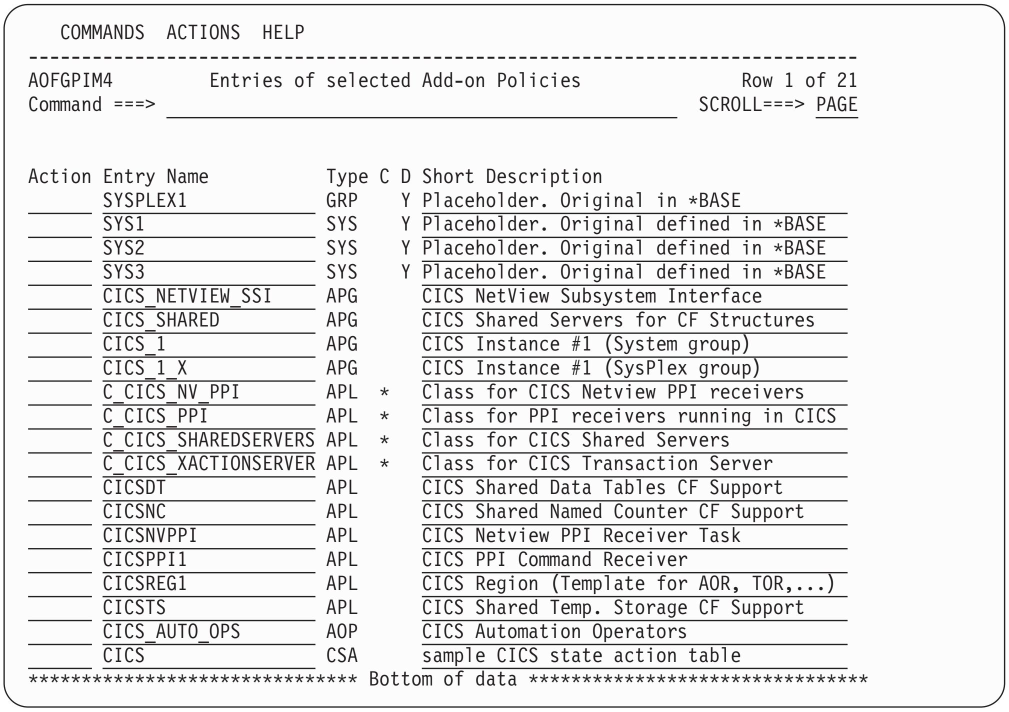


```
   ACTIONS   HELP
------------------------------------------------------------------------------
AOFGPIMA                    Import Add-on Policies               Row 1 of 15
Command ===> ________________________________________________

1 Import selected add-on policies
2 View import report

Current Policy Database: BASE
Add-on policies to be added to the current policy database:
Action     Status      Add-on Policy         Customizable
_________              *CICS
_________              *DB2
_________              *E2E
_________              *GDPS
_________              *HYPERSWAP
_________              *IBMCOMP              YES
_________              *IMS
_________              *ITM                  YES
_________              *NMC
_________              *PROCOPS
_________              *SAP                  YES
_________              *TBSM
_________              *TWS
_________              *USS
_________              *WEBSPHERE            YES
****************************** Bottom of data *******************************
```

Figure 266. Import Add-On Policies Panel

3. Select the add-on samples that you want to import and invoke option 1 (Import selected add-on policies).

   This performs a check of the selected policies. Then a panel similar to the one shown in Figure 267 is displayed.



```
   COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGPIM4         Entries of selected Add-on Policies              Row 1 of 21
Command ===> ________________________________________________  SCROLL===> PAGE


Action Entry Name          Type C D Short Description
______ SYSPLEX1             GRP    Y Placeholder. Original in *BASE
______ SYS1                 SYS    Y Placeholder. Original defined in *BASE
______ SYS2                 SYS    Y Placeholder. Original defined in *BASE
______ SYS3                 SYS    Y Placeholder. Original defined in *BASE
______ CICS_NETVIEW_SSI     APG      CICS NetView Subsystem Interface
______ CICS_SHARED          APG      CICS Shared Servers for CF Structures
______ CICS_1               APG      CICS Instance #1 (System group)
______ CICS_1_X             APG      CICS Instance #1 (SysPlex group)
______ C_CICS_NV_PPI        APL  *   Class for CICS Netview PPI receivers
______ C_CICS_PPI           APL  *   Class for PPI receivers running in CICS
______ C_CICS_SHAREDSERVERS APL  *   Class for CICS Shared Servers
______ C_CICS_XACTIONSERVER APL  *   Class for CICS Transaction Server
______ CICSDT               APL      CICS Shared Data Tables CF Support
______ CICSNC               APL      CICS Shared Named Counter CF Support
______ CICSNVPPI            APL      CICS Netview PPI Receiver Task
______ CICSPPI1             APL      CICS PPI Command Receiver
______ CICSREG1             APL      CICS Region (Template for AOR, TOR,...)
______ CICSTS               APL      CICS Shared Temp. Storage CF Support
______ CICS_AUTO_OPS        AOP      CICS Automation Operators
______ CICS                 CSA      sample CICS state action table
****************************** Bottom of data *******************************
```

Figure 267. Entries of Selected Add-on Policies Panel

4. The Entries of selected Add-on Policies panel allows you to modify the entries of the selected add-on policies for the import. You can:

- Press Enter if the list is okay.
- Remove entries from the list by blanking their names or entering REMOVE(M) in the Action field
- Rename entries in the list by overtyping their names or entering RENAME(R) in the Action field

  If an entry of same name and type already exists in the target policy database, it will be marked as duplicate (D is set to “Y”). If you change a duplicate name to a non-duplicate one, column D will be reset to blank. Duplicate entries will not be imported.
- Overtype the entry description field (effects non-duplicate entries only).
- Locate an entry using LOCATE(L) on the command line.
- Return to the previous panel by pressing END or entering the CANCEL command.

Note: If the entry is a class, column C will be set to “*”.

When you are done with modifications and have pressed Enter, a command progress display is shown. After the import has successfully finished, you will be taken back to the Import Add-On Policies panel shown in Figure 266 on page 355.

5. If you want to check what happened, invoke option 2 (View import report) on the Import Add-On Policies panel.

Warning: Consistent Naming

Importing an add-on policy includes importing all relationships and links between policy objects in that database.

Therefore, if you start either from a copy of the `*BASE` sample policy database, but modify it before importing an add-on, or from a policy database that you have created yourself on the basis of the `*EMPTY` sample policy database, you need to make sure that naming is consistent across your current policy database and the sample that you want to add on. This is so that relationships and links can be resolved during the import operation. When there are relationships or links that cannot be resolved, an “unresolved link” message is displayed in a progress indication panel.

Making Updates with Text Files

You can carry out bulk updates of policy objects using a flat text file. You can select one or more policy objects and their associated policy items and have their data written to a text file. You can then edit the text file to update the policies and then import the updated file into your policy database.

This update method can be quicker than using the customization dialog if you want to update several entries, but you must take care that the syntax is correct.

Currently, only policy objects of type APG, APL, MTR, and UET are supported.

Note that the flat-file bulk update facility does not allow concurrent user access.

For example, to update APL policy objects via a text file, follow these steps:

Step 1. **Edit the text file:**

Use option 5 (Edit output file) on the Policy Data Base Update Selection panel to edit the data set where you put your text policy data.

Step 2. **Check syntax (optional):**

Although a syntax check is performed when you import a text file, you can use option 2 (Perform syntax check) to check that the syntax of the file data complies with the data in the policy database before you import the file. This can save you time because it allows you to check the syntax without invoking the complete import process.

Step 3. **Import the updated text file:**

On the Data Management Menu panel, make sure that the **Input File Name** field specifies the data set with your updated policy data. If you do not provide a name in this field, SA z/OS will put in the name of the data set previously used for output.

Type 3 on the option line and press Enter.

Further update processing is controlled by the **Mode** setting:

ONLINE

This is the default value. It indicates that the update should run in the foreground. It displays a command progress display. Processing is as follows:

a. A syntax check of the whole text file is performed. Only when the syntax check has finished successfully does the actual policy database update take place.

b. If there is a timestamp in the text file and it is more recent than the last modification of entries in the policy database, the policy database is modified.

   Otherwise a confirmation panel with the names of the most recently modified entries is displayed, where you can either confirm or cancel the update processing.

Messages from the update are written to a report data set that can be viewed with option 4 in the Policy Database Update Selection panel.

The data set with the text policy data is not affected by the import. In particular it is not deleted.

BATCH

This indicates that the update should run in the background. Processing is as follows:

a. A batch job is started, executing the policy database update as soon as it obtains exclusive access to the database. This requires you to specify valid JOB statement information in the panel.

b. A syntax check of the whole text file is performed. Only when the syntax check has finished successfully, does the actual policy database update take place.

c. If there is a timestamp in the text file and it is more recent than the last modification of entries the policy database, the policy database is modified.

   Otherwise the update process ends with messages in the report data set giving the reasons for ending without updating the policy database.

The settings in the **Mode** field and **Job statement information** are saved in your ISPF profile data set and are used as the initial values the next time you issue the UPDATE command.

Supported Policies

Besides the policies that are available for applications, the following policies are supported for other entry types:

Application Groups
DESCRIPTION
APPLGROUP INFO
RELATIONSHIPS
MESSAGES/USER DATA

Monitor Resources
DESCRIPTION
MONITOR INFO
RELATIONSHIPS
MESSAGES/USER DATA
HEALTHSTATE

User E-T Pairs
DESCRIPTION
E-T DATA

Updating Policy Text Files

Each policy object in the text file is represented by the following structure:

```
UPD <Entry Type>       : <Entry Name>
  <Policy Item>         ( field|block|policy replacement )
    <Field>             : <Value>
    <Field>             : <Value>
    ...
  <Policy Item>         ( field|block|policy replacement )
    <Field>             : <Value>
    <Field>             : <Value>
    ...
  ...
```

The actual values for the Entry Type, Entry Name, Policy Item and Field variables depend on the selections that you made when the file was created and what data is in your policy database, for example:

```
UPD APL                        - AM
  RELATIONSHIPS                ( policy replacement )
    Relationship Type          : MAKEUNAVAILABLE
    Supporting Resource        : AM2/APL/=
    Description                : Prevents takeover at system shutdown
    Sequence Number            :
    Automation                 : PASSIVE
    Chaining                   : WEAK
    Condition                  : WhenDown
* MESSAGES/USER DATA             no data available
```

Do not modify the *<Policy Item>* or *<Field>* segments because the syntax of the file will be invalid and the file will fail syntax checking.

If the file does fail the syntax check, you can view the update report and fix any errors that are reported before trying the update again.

The text after each policy item indicates which type of data replacement is possible:

- Field replacement: the data that you specify in the text file replaces only the corresponding data in the policy database and nothing else.
- Block replacement: the row and all data that is nested underneath it is treated as one block of data, and each such block replaces the corresponding data in the policy database.
- Policy replacement: the policy definition in the text file replaces the whole policy definition in the policy database.

Note: The <*Policy Item*> lines must begin in column 2 or 3 or the file will fail the syntax check.

Field Replacement

With *field replacement* the specified data in the file replaces only the corresponding data in the policy database. A blank value for a field in the file discards a value that may be in the policy database or resets it to a default in the same way as for the input field on the panel. The data in any other fields that belong to a policy but are not specified in the text file remains unchanged.

Block Replacement

With *block replacement* a data row marked with a dash indicates a data block that consists of that row and all data that is nested underneath it. Each such block replaces the corresponding data in the policy database. Other blocks for the same policy item remain unchanged. A data block where you have edited all the fields to blank would delete all the corresponding data from the policy database. For example, the following would delete the reply definitions from a message:

```
Action                  - REPLY
        Pass/Selection        :
        Retry Count           :
        Reply Text            :
```

You can also specify just the block header line to delete all the data in that block. Thus, the equivalent for previous example would be:

```
Action                  - REPLY
```

Example 1: Create or Change Policy Messages for APL TSO in the *BASE Policy Database: All of these changes are made in one file.

1. The data in the text file is initially:

   ```
   UPD APL                        - TSO
     MESSAGES/USER DATA           ( block replacement )
       Message id                 : IKT001D
         Description              : Active users during TCAS start
         Action                   - REPLY
           Pass/Selection         :
           Retry Count            :
           Reply Text             : U
   ```

   Change Message ID IKT001D to TEBULK and create a new message for this message ID:

   ```
   UPD APL                        - TSO
     MESSAGES/USER DATA           ( block replacement )
       Message id                 : TEBULK
         Description              : Active users during TCAS start
         Action                   - REPLY
           Pass/Selection         :
           Retry Count            :
   ```

```
        Reply Text             : U
        Pass/Selection         : *
        Retry Count            : 1
        Reply Text             : dump
```

2. Create a new Message ID TEST for the application TSO:

```
UPD APL                        - TSO
  MESSAGES/USER DATA           ( block replacement )
...
    Message id                 : TEST
      Description              : test bulk
      Action                   - code
        Code 1                 : 9999
        Code 2                 :
        Code 3                 :
        Value Returned         : TERM
```

Example 2: Change or Insert SHUTDOWN for the APL C_TWS in Sample *TWS Policy Database:

1. Delete pass 4 for PHASE NORM, change the pass from 2 to 4 for PHASE IMMED, and insert a reply for PHASE INIT:

```
UPD APL                        - C_TWS
  SHUTDOWN                     ( block replacement )
    Phase                      : NORM
      Action                   - CMD
        Pass                   : 1
        Automated Function/'*' :
        Command Text           : MVS P &SUBSJOB
        Pass                   : 4
        Automated Function/'*' :
        Command Text           : MVS C &SUBSJOB
    Phase                      : IMMED
      Action                   - CMD
        Pass                   : 1
        Automated Function/'*' :
        Command Text           : MVS P &SUBSJOB
        Pass                   : 4
        Automated Function/'*' :
        Command Text           : MVS C &SUBSJOB
    Phase                      : FORCE
      Action                   - CMD
        Pass                   : 1
        Automated Function/'*' :
        Command Text           : MVS C &SUBSJOB
    Phase                      : INIT
      Action                   - REPLY
        Reply Text             : TEST BULKUPDATE
```

Policy Replacement

With *policy replacement* the data in the text file replaces the whole policy item definition in the policy database. For example:

- An application has two relationships defined for it in the policy database and one relationship in the text file. After the file update has been completed the application has only the one relationship from the text file.
- An application has some USS data defined in the policy database, while all USS fields in the text file for the application are blank. After the file update has been completed the application no longer has any USS data.

Empty lines are allowed anywhere in the file. They are ignored when the file update is performed. Comment lines are also allowed anywhere in the file. They have an asterisk in the first column and are skipped during file import.

If a field is specified several times, the last definition wins, even if it is an empty field. The following sequence in the APPLICATION INFO policy would remove any Owner data from the policy database for the corresponding application:

```
Owner               : OPERATOR
    Owner               :
```

Example: Create or Change Policy Relationships for the APLs TCPIP, JES2 and JES3 in the *BASE Policy Database: All of these changes are made in one file.

1. The data in the text file is initially:

   ```
   UPD APL                     - JES2
     RELATIONSHIPS             ( policy replacement )
       Relationship Type       : HASMONITOR
       Supporting Resource     : JES2MON/MTR/=
       Description             : Relationship to JES2 Monitor
       Sequence Number         :
       Automation              :
       Chaining                :
       Condition               :
   *
       Relationship Type       : PREPUNAVAILABLE
       Supporting Resource     : VTAM/APL/=
       Description             : Delay DRAINJES until VTAM is down
       Sequence Number         :
       Automation              : PASSIVE
       Chaining                : WEAK
       Condition               : WhenDown
   ```

   Change the second relationship for JES2:

   ```
   UPD APL                     - JES2
     RELATIONSHIPS             ( policy replacement )
       Relationship Type       : HASMONITOR
       Supporting Resource     : JES2MON/MTR/=
       Description             : Relationship to JES2 Monitor
       Sequence Number         :
       Automation              :
       Chaining                :
       Condition               :
   *
       Relationship Type       : PREPAVAILABLE
       Supporting Resource     : VTAM/APL/=
       Description             : Edit the description for test
       Sequence Number         : 1
       Automation              : PASSIVE
       Chaining                : WEAK
       Condition               : WhenDown
   ```

2. Add a new relationship to JES3. The data in the text file is initially:

   ```
   UPD APL                     - JES3
   * RELATIONSHIPS               no data available
   ```

   Edit the file as follows:

   ```
   ...
   UPD APL                     - JES3
     RELATIONSHIPS             ( policy replacement )
       Relationship Type       : HASMONITOR
       Supporting Resource     : JES3MON/MTR/=
       Description             : Relationship to JES3 Monitor
       Sequence Number         :
       Automation              :
       Chaining                :
       Condition               :
   ```

3. Delete the second relationship from TCPIP (blank all fields or delete all lines for the second relationship). The data in the text file is initially:

```
UPD APL                     - TCPIP
 RELATIONSHIPS              ( policy replacement )
  Relationship Type         : HASPARENT
  Supporting Resource       : RESOLVER/APL/=
  Description               : TCPIP requires RESOLVER for addr xlation
  Sequence Number           :
  Automation                :
  Chaining                  :
  Condition                 :
*
  Relationship Type         : HASPARENT
  Supporting Resource       : VTAM/APL/=
  Description               : TCPIP is dependent on VTAM
  Sequence Number           :
  Automation                :
  Chaining                  :
  Condition                 :
```

Edit the file, for example, as follows:

```
...
UPD APL                     - TCPIP
 RELATIONSHIPS              ( policy replacement )
  Relationship Type         : HASPARENT
  Supporting Resource       : RESOLVER/APL/=
  Description               : TCPIP requires RESOLVER for addr xlation
  Sequence Number           :
  Automation                :
  Chaining                  :
  Condition                 :
*
  Relationship Type         :
  Supporting Resource       :
  Description               :
  Sequence Number           :
  Automation                :
  Chaining                  :
  Condition                 :
```

Creating New Policy Objects via File Updates

To create new policy objects, insert sections that have the following structure:

```
NEW <Entry Type>               : <new entry name>
COPY                           : <name of existing entry to copy data from>
WHERE_USED <parent entry type> : <name of existing entry to link to>
LINK_TO_CLASS                  : <name of existing APL class entry to link to>
```

You can create new policy objects for any entry type *except* for type ENT (enterprise).

These sections can be inserted anywhere in the text file, with the following conditions:

- Policy object sections are not broken apart by other ones.
- The keywords (NEW, COPY, WHERE_USED, and LINK_TO_CLASS) *must* begin in column 1. Any <*Policy Item*> lines must begin in column 2 or 3. All other data must be indented.

The keywords LINK_TO_CLASS (for entry type APL only), COPY, and WHERE_USED are optional. If they are specified, they can appear in any order.

The NEW keyword allows additional specifications, for example, subsystem name, in the same notation as for the update function. All the fields that are available on the NEW panel for the entry type that you are creating are also valid here. For detailed examples, see Appendix C, “Example New Policy Objects via File Updates,” on page 403.

The COPY keyword works in the same way as the COPY policy in the customization dialog panels. This means that the same limitations apply with respect to type, subtype, or class/instance compatibilities.

For each keyword, the specification of only one value is allowed. To link an application to several application groups, the WHERE_USED keyword may appear as many times as you require.

The WHERE_USED keyword may need an optional parameter, <*parent entry type*>, for entries that might be linked to several parent types, for example, Service Periods to Applications, Application Groups or Monitor Resources.

If the parent entry type is omitted, the following defaults are assumed:

- For SYS the default is GRP
- For SVP the default is APL
- For TRG the default is APL.

The parent entry type for Application Groups defaults according to the Application Group’s type. If the type is SYSPLEX the default link parent entry type is Group, otherwise it is System.

If either of the COPY or LINK_TO_CLASS keywords is specified several times, the last one is used (the last specification overwrites any previous one) even if it is blank.

The following example shows a more elaborate definition:

```
NEW APL             : $IMS711_MSG_REGION5
  Subsystem Name    : IMS711M5
  Application Type  : IMS
  Job Name          : IMS711M5
  Short description : IMS V7 - IMS711 Message region on SYS1
LINK_TO_CLASS       : CLASS_IMS_TP
WHERE_USED APG      : IMS711_CONTROL
WHERE_USED APG      : APPLGR_MAINT
COPY                : IMS713M0
```

Note: You should take care with the order of your definitions. It is possible that a class link that you set with a LINK_TO_CLASS definition is overwritten by a class link that is inherited by a COPY command that is executed after the LINK_TO_CLASS definition.

The instructions to create new entries and the data to modify existing fields are both run from a flat file. As a consequence they can be combined so that the following would be possible:

```
NEW APL      : IMS721RC
COPY         : IMS713RC
WHERE_USED   : IMS721_CONTROL
LINK_TO_CLASS: CLASS_IMS_DBRC

UPD APL                 : IMS721RC
  DESCRIPTION
    Short description   : IMS V7 - IMS721 DBRC region on SYS2
  RELATIONSHIPS
    Relationship Type   : HASPARENT
```

```
Supporting Resource : IMS721C4/APL/=
Description         :
Sequence Number     : 1
Automation          :
Chaining            :
Condition           :
```

If the keyword NEW or UPD is specified more than once with the same application name, only the last specification is processed. All previous are ignored. If a specification is ignored, all data rows are skipped up to the next occurrence of NEW or UPD keyword. If a NEW or UPD keyword is skipped, an informational message is issued.

Generating a Job for Batch Update

You can run a job for batch update that performs the same processing as the UPDATE function of the customization dialog. The SINGSAMP sample library contains a sample job for this task in member INGEBFLT. Before submitting this job, you need to edit the jobcard appropriately as described in the sample itself.

The batch update produces the following return codes:

0 Successful

4 Syntax errors or timestamp conflicts were detected; the update was terminated. See the job log or report file for details.

8 The update text file could not be read or the report data set could not be reallocated. The update was terminated.

>8 Severe error; update terminated. See the job log for details.

Migrating the Automation Control File

Note:
This section applies to migrating an automation control file for SA z/OS 3.1 enterprises or earlier only.

For the automation manager, it is necessary to create the automation manager configuration file, which can only be built using the customization dialog. Thus all data must be available in the policy database.

If the customization dialog has not yet been used, and all automation definitions are stored in an ACF that has been edited manually, the MIGRATE function of the customization dialog should be used. It recognizes all valid ACF keywords, and merges the corresponding values into an existing policy database. In detail, the following steps are required:

1. If the customization dialog has never been used before, a user-specific policy database does not exist. To create one, specify **option 4** on the Customization Dialog Primary Menu to go to the Policy Database Selection panel, and enter the command `NEW`.

   On the subsequent panel, a model policy database (PolicyDB) name is required. Entering a question mark in the corresponding field gives a list of samples where one can be selected. A description of the samples can be found in "Using SA z/OS Sample Policies" on page 323. This allows you to select a sample that best matches your installation requirements.

If a new policy database has been created, it will be listed on the Policy Database Selection panel.

2. On the Policy Database Selection panel, use the Data Management command, MANAGE (G), for the policy database that you want to copy the automation control file data to.

   The Data Management Select panel is displayed. On this panel, specify the entry name for the system that will be used as the target and the MVS system name. It is not necessary that this system already exists. If it does not exist, the MIGRATE function will create the system automatically.

   In the **Source Data Set** field, enter the data set name and the member name that contains the ACF definitions (or the main member containing the include statements for the fragments) for the system to be migrated.

   If you have several systems defined, each with its own ACF, they have to be migrated one after the other. It is not possible to migrate multiple systems at the same time, because a specific target system must always be specified.

3. Before the migration is started, specific objects can be selected or excluded from migration. However, if a complete ACF is to be migrated into a new policy database, it is not necessary to go into these details. By default, all entry types are selected, and the migration can be started without any further specification. The MIGRATE function will copy all source definitions into the corresponding entry types automatically.

4. If it is necessary to select specific objects for migration, the corresponding option on the Data Management Select panel has to be selected. This gives a list of all entry types found in the source data set. There each entry type can be selected or excluded for migration. And if it is necessary to be even more specific, for each entry type a list of the entry names found for that type can be displayed. There it is possible to check for duplicates, and rename entries, if required.

   If a complete ACF is migrated into a new policy database, this selection of specific entry types and names can be skipped.

5. If the preparations are done, the migration itself can be started. Because it may take some time to complete, the progress is indicated on a special screen that is updated automatically and shows all the migration steps as they are executed. Furthermore, a report of the migration is written and stored as member in the target data set. The member name needs to be specified on the Data Management Select panel.

6. All data belonging to one entry in the ACF are copied into one entry within the target policy database. Depending on the specification in the ACF, it may be more or less difficult for the migration to find the name of such an entry.

7. Any lowercase characters encountered in the automation control files that are being migrated will be translated to uppercase unless they are enclosed in matching pairs of either ″ or ′.

Note:

Migration of EVT, MTR, SVP, and TRG resources is not supported.

You have the following alternative methods of invoking the migration function:

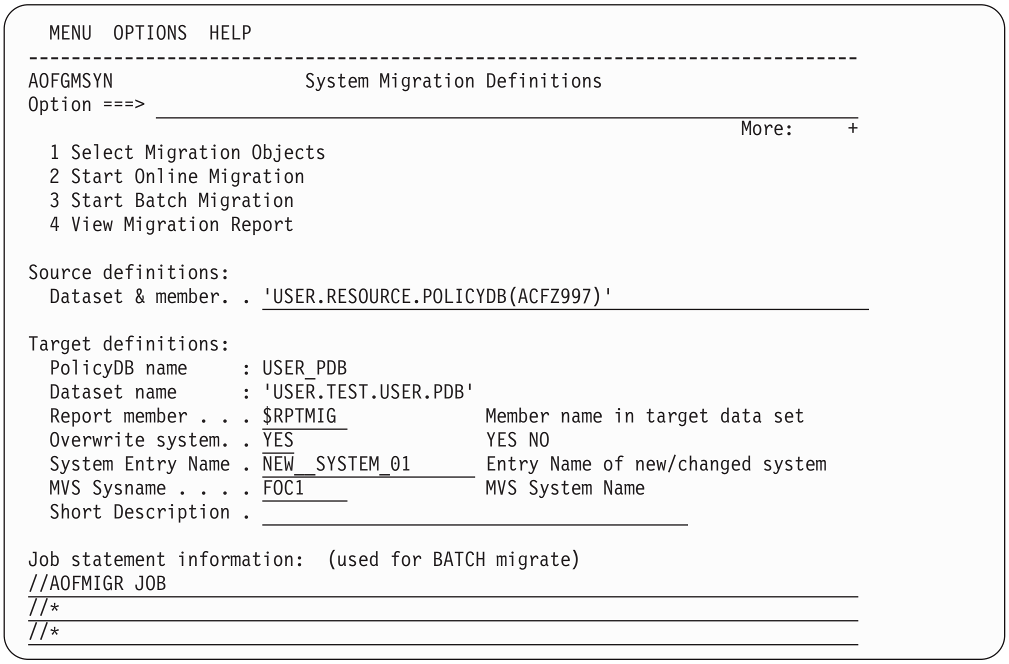
- From the Customization Dialog Primary Menu (Figure 18 on page 34) select option *5 Data Management*.
- From the Policy Database Selection panel, you can select action *Migrate (G)* against the policy database to that you want to migrate your data to.

In both cases, you reach the System Migration Definitions panel shown in Figure 261 on page 351 that you can use to migrate an automation control file into a policy database of the current release of SA z/OS

For every enterprise (every policy database) you migrate at a System level, that is you migrate each single automation control file main include fragment for each system.

Note:
The more selectively you specify your migration process the better the performance will be. For example, specifying a certain automation control file fragment instead of specifying the main include member may be useful if you only want to migrate certain parts of an enterprise. The various selection methods that are available are described below.

Selecting the migrate option leads to the panel shown in Figure 268.



```
  MENU  OPTIONS  HELP
------------------------------------------------------------------------------
AOFGMSYN                  System Migration Definitions
Option ===>
                                                                   More:     +
  1 Select Migration Objects
  2 Start Online Migration
  3 Start Batch Migration
  4 View Migration Report

Source definitions:
  Dataset & member. . 'USER.RESOURCE.POLICYDB(ACFZ997)'

Target definitions:
  PolicyDB name     : USER_PDB
  Dataset name      : 'USER.TEST.USER.PDB'
  Report member . . . $RPTMIG              Member name in target data set
  Overwrite system. . YES                  YES NO
  System Entry Name . NEW__SYSTEM_01       Entry Name of new/changed system
  MVS Sysname . . . . FOC1                 MVS System Name
  Short Description .

Job statement information:  (used for BATCH migrate)
//AOFMIGR JOB
//*
//*
```

Figure 268. System Migration Definitions Panel

Note:
Only ACF data from the following releases can be used for migration:

- SA z/OS 3.1
- SA z/OS 2.3
- SA OS/390 2.2
- SA OS/390 2.1

ACF data from a release later than SA z/OS 3.1 will be rejected.

This panel allows you to specify the information required to migrate a system's data from an automation control file fragment into a target policy database. The system name that you specify in the **System Entry Name** field is either created in the target policy database or, if it already exists, it is overwritten. You have the following options:

- **1 Select Migration Objects**

  This option allows you to specify the selective migration of objects from ACF fragments. It leads to the System Object Migration panel (Figure 269 on page 369) where you can select those entry types that you want to migrate to the current policy database whilst explicitly excluding the rest of the inputs.

  Within the selected entry types you can later select certain existing entry names that you want to migrate. To achieve this, enter the command EDIT (E) against a selected entry type. The Entry Name Migration panel (Figure 270 on page 370) is then displayed.

  After you have selected all objects (entry types and entry names) for migration, return to the System Migration Definitions panel where you can now start the migration using one of the two following Start Migration options.

- **2 Start Online Migration**

  This option immediately starts the migration of the specified ACF fragment to the current policy database in the foreground.

- **3 Start Batch Migration**

  This option starts the migration process in the background. In this case, you need to provide a valid JOB statement in the **Job statement information** field on the same panel.

  For definitions of the data set allocations that are needed for batch migration, see “Defining Data Sets for Batch Processing” on page 396.

- **4 View Migration Report**

  This option invokes the ISPF VIEW function to view the migration report that was created as specified in the **Report member** field for this target policy data set.

Notes:

1. If option 1, 2, or 3 is entered the *first* time, all ACF data is read and saved for further processing *until you exit this panel*.
2. If option 2 or 3 is used after option 1 has been used, and the selections have been saved, this data is used for migration.
3. If the selections have been canceled, processing starts again by reading the ACF data.
4. If option 1 is used again after the selections have been canceled, processing starts again by reading the ACF data.

In the fields below these options, specify the following source and target definitions:

Dataset & member
This specifies the name of a partitioned data set together with the member name that contains the ACF of the system to be migrated, for example, `'ADMIN1.FOCAL.LIB(ACFZ998)'`. The member name must be specified. The source data set must not be the same as the target data set.

PolicyDB name
Already specified and cannot be changed.

Dataset name
Already specified and cannot be changed.

Report member
This specifies the member name where the migration report will be written. The member will be created in the TARGET data set.

Overwrite System
This specifies whether the system entry in the policy database will be overwritten during migration if the system entry already exists. For existing system entries you have to specify YES to continue.

> **Note:**
> In this field, you can prevent SA z/OS from unintentionally overwriting existing information in the policy database for this system. The default is NO. A message, `System entry name exists`, is displayed if this field is set to NO and you try to use the same system entry name as already exists in the policy database. To proceed, you must either change the system name or set this field to YES.

System Entry Name
This specifies the entry name of the system to be migrated to. Note especially that you also define the system name that you want to migrate the old system's ACF fragment to.

> **Note:**
> If the system already exists in the target policy database it will be updated with the migration data, otherwise the new system will be created. To verify that the system already exists press Enter

MVS Sysname
This specifies the MVS system identifier for the migrated system.

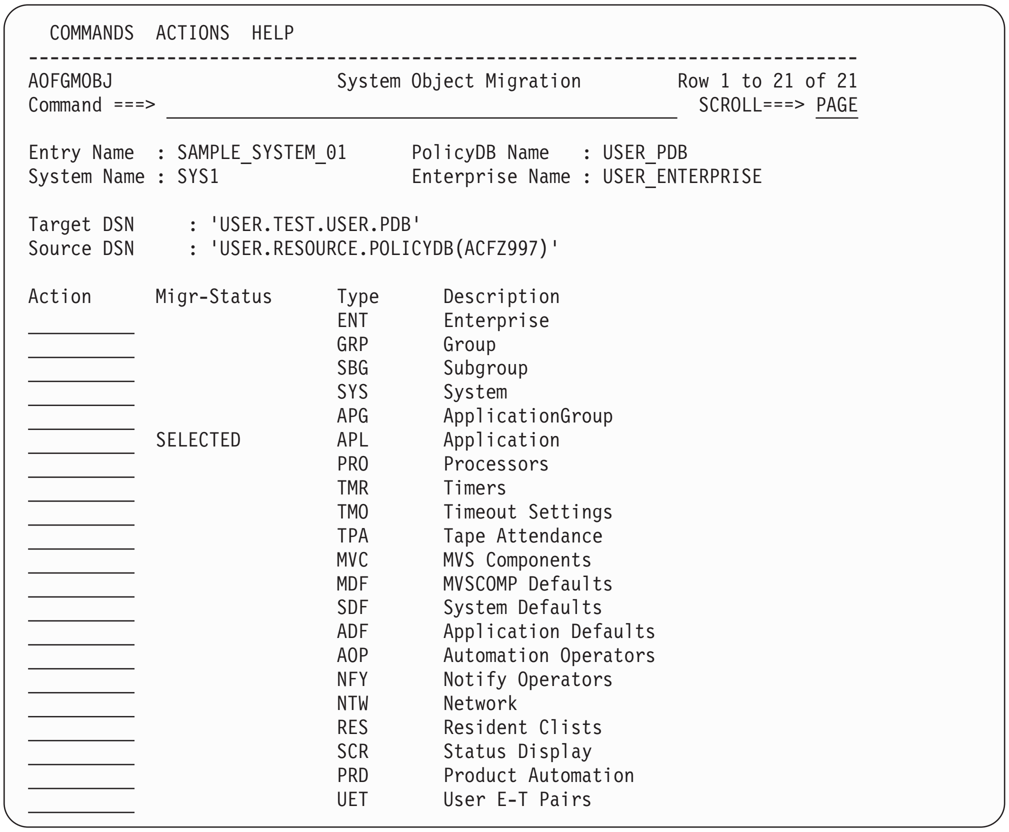
> **Note:**
> If the system already exists in the target policy database the MVS SYSNAME must be the one already defined in the policy database. The special notation with a trailing 'plus-sign' is allowed. The first three characters of the SYSNAME must not be AOF because these are reserved for internal processing of the SA z/OS System Operations functions.

Short Description
This specifies the short description of the migrated system.

If you specified that the MIGRATE request should be run in BATCH mode then you need to provide a valid JOB statement in the **Job statement information** field.

The information that you specify here is saved in your ISPF profile data set and used as the initial values for the next MIGRATE request.

Selecting option **1 Select Migration Objects** on the System Migration Definitions panel leads to the panel shown in Figure 269.



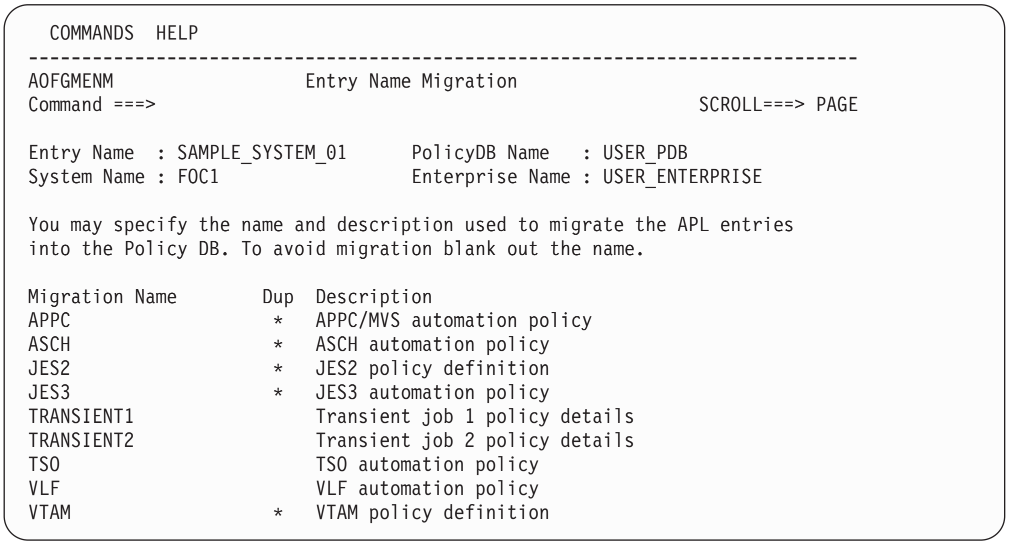
```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGMOBJ                    System Object Migration        Row 1 to 21 of 21
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Name  : SAMPLE_SYSTEM_01      PolicyDB Name   : USER_PDB
System Name : SYS1                  Enterprise Name : USER_ENTERPRISE

Target DSN     : 'USER.TEST.USER.PDB'
Source DSN     : 'USER.RESOURCE.POLICYDB(ACFZ997)'

Action      Migr-Status      Type      Description
                             ENT       Enterprise
__________                   GRP       Group
__________                   SBG       Subgroup
__________                   SYS       System
__________                   APG       ApplicationGroup
__________  SELECTED         APL       Application
__________                   PRO       Processors
__________                   TMR       Timers
__________                   TMO       Timeout Settings
__________                   TPA       Tape Attendance
__________                   MVC       MVS Components
__________                   MDF       MVSCOMP Defaults
__________                   SDF       System Defaults
__________                   ADF       Application Defaults
__________                   AOP       Automation Operators
__________                   NFY       Notify Operators
__________                   NTW       Network
__________                   RES       Resident Clists
__________                   SCR       Status Display
__________                   PRD       Product Automation
__________                   UET       User E-T Pairs
__________
```

Figure 269. System Object Migration Panel

All entry types are initially selected on this panel. You can issue the following actions for the entry types:

- **ADD (A)** to add an entry type to the SELECTED list; if you type `A *` into the command line, all entry types will be selected.
- **REMOVE (M)** to remove an entry type from the SELECTED list; if you type `M *` into the command line, all selected entry types will be deselected.
- **EDIT (E)** to edit the selected entry type before migration; you will be presented with the Entry Name Migration panel with a list of entries of that type where you can select the ones you want to migrate or blank out the names of those that you do not want to migrate (see Figure 270 on page 370).

If you enter the command CANCEL, all changes made on this panel and on the Entry Name Migration panel (which is displayed if action ″EDIT″ was selected) are deleted and the migration process starts as if it has been invoked for the first time.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGMENM                  Entry Name Migration
Command ===>                                                  SCROLL===> PAGE

Entry Name  : SAMPLE_SYSTEM_01      PolicyDB Name   : USER_PDB
System Name : FOC1                  Enterprise Name : USER_ENTERPRISE

You may specify the name and description used to migrate the APL entries
into the Policy DB. To avoid migration blank out the name.

Migration Name        Dup  Description
APPC                   *   APPC/MVS automation policy
ASCH                   *   ASCH automation policy
JES2                   *   JES2 policy definition
JES3                   *   JES3 automation policy
TRANSIENT1                 Transient job 1 policy details
TRANSIENT2                 Transient job 2 policy details
TSO                        TSO automation policy
VLF                        VLF automation policy
VTAM                   *   VTAM policy definition
```

Figure 270. Entry Name Migration Panel

An asterisk (*) in the **Dup** column indicates that the specified name already exists in the policy database. The existing entry will be replaced during migration.

Remember:
You can only start migration from the System Migration Definitions panel using either option **2 Start Online Migration** or **3 Start Batch Migration**.

When you have finished migrating your existing automation control file to the selected policy database, you need to use the BUILD command to build a new automation control file.

Notes:

1. Migrating an automation control file does not build an SA z/OS automation control file. The information from your old automation control file is migrated into an SA z/OS policy database. To build an SA z/OS automation control file, you must use the Build Functions Menu.
2. If you have automation control files from multiple systems to migrate, migrate each automation control file into your policy database for each system.

Entry Name Creation

Table 9 describes how SA z/OS creates names and descriptions for the entry types to be migrated.

Table 9. Creation of Entry Names

Entry Type	Entry Name	Description
ENT	From table AOFTENT (ENTNAME)	From table AOFTENT (ENTDESCS)
SYS	From panel AOFGMSYN	From panel AOFGMSYN

Table 9. Creation of Entry Names (continued)

Entry Type	Entry Name	Description
GRP	From ACF line ’GROUP entryname’	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
SBG	From ACF line ’SUBGROUP entryname’	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
APG	From ACF line ’APPLGROUP entryname’	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
APL	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or from ACF line ’SUBSYSTEM_sysname subsysname’ if the ’* Built’ line is not available. The name created will be: ’subsysname_sysname’.	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
PRO	From ACF line ’PROCESSOR entryname’	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
TMR	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’TIMER_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
TMO	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’TMO_SET_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
TPA	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’TAPE_ATT_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
MVC	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’MVS_COMP_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
MDF	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’MVSCOMP_DFT_sysname’ if the ’* Build’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
SDF	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’SYSTEM_DFT_sysname’ if the ’* Build’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
ADF	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’APPL_DFT_sysname’ if the ’* Build’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
AOP	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’AUTOOPS_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available

Table 9. Creation of Entry Names (continued)

Entry Type	Entry Name	Description
NFY	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’NOTIFYOP_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
NTW	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’NETWORK_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
NNT	Fragments of the type NNT are no longer accepted. If data is migrated via another include member, the data is treated as user E-T pairs.	
RES	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’RES_CLISTS_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
SCR	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’STAT_DETAIL_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available
PRD	not applicable	not applicable
UET	From ACF line ’* Built : userId yy/mm/dd hh:mm:ss entryname’ or ’USER_ET_sysname’ if the ’* Built’ line is not available	From ACF line ’* Desc : short_description’ or ’’ if the ’* Desc’ line is not available

The MIGRATE function checks for the comment line starting with ** Built*, and takes the entry name from there. Such a line is created automatically, if the ACF is built using the customization dialog.

If such a comment line is not available, a fixed entry name is created as described in Table 9 on page 370 and all definitions are stored into this entry. In a manually edited ACF, it may be helpful to insert comment lines with entry names as shown above to group the definitions accordingly, so that the migration is able to store the definitions into the policy database with their meaningful names.

Restrictions: If an application or an application group entry name is migrated into the policy database and the same entry name already exists, only the existing subsystem/automation name will be used during the migration process. The subsystem/automation name from the input will be ignored. The reason for this is that these names are used for creating resource names and when the subsystem name or automation name has changed, the resource name would be invalidated and the automation manager data would be incorrect afterwards.

Notes:

1. Definitions relevant for subsystems need to be migrated *together* with the subsystem's ACF data *in one step* in order to have them migrated to the correct application. If this is not observed, the definitions will be migrated to User E-T Pairss.
2. If migrating Application fragments, an application group is automatically created. This is required to guarantee that the application is linked to the appropriate system. All applications found in the migration ACF-source are linked to this application group. The entry name of this application group is *APG_sysname*.
3. If no description has been given in the entry name migration panel (AOFGMENM), the description found in the ACF fragment (keyword SDESC or DESC) will be used.

Generating a Job for Batch Migrate

You can run a job for batch migration that performs the same processing as the Migration function of the customization dialog. The SINGSAMP sample library contains a sample job for this task in member INGEBMIG. Before submitting this job, you need to edit the jobcard appropriately as described in the sample itself.

Chapter 10. Automating Networks

Automation Network Definition Process

This section summarizes the steps for defining an automation network to SA z/OS. More detail for each step of the process is provided later in this chapter.

1. Define your message forwarding paths between different systems. To do this, you define:
   - A primary focal point, where all notifications are sent.
   - An optional backup focal point, used when the primary focal point is unavailable.
   - Target systems, which are monitored and controlled by the focal point system.
   - Gateway sessions between the systems.

     "Defining Gateway Sessions" on page 377 describes how to define gateway sessions.
2. Modify the NetView definitions to reflect your automation network configuration. The chapter on how to install SA z/OS on host systems in *IBM Tivoli System Automation for z/OS Planning and Installation* provides details.

For an example of the automation network definition process, also refer to the chapter about installing SA z/OS on host systems in *IBM Tivoli System Automation for z/OS Planning and Installation*.

These definitions create a path allowing message forwarding from target systems to the focal point system.

A message forwarding path is best implemented by defining systems in the following top-down manner:

1. Primary focal point system
2. Backup focal point system
3. Target systems

Defining the primary focal point first ensures that it is ready to handle forwarded messages as soon as forwarding is turned on for the target systems.

If the message forwarding path is not yet implemented on all systems in an automation network, messages are displayed to notification operators on the target systems. Once the message forwarding path is implemented, notifications are forwarded to the focal point system.

If the target systems are implemented first, additional overhead occurs because the target systems unsuccessfully attempt to forward notifications, and the notifications are logged in the NetView log.

Defining an SDF Focal Point System

The focal point system and backup focal point systems are defined using the Network entry type in the customization dialog. Each system has a single entry in the automation policy defining the next system or domain in the message forwarding path. Figure 271 shows an example automation network. In this example, the primary focal point system is CHI01. The backup focal point is CHI02.

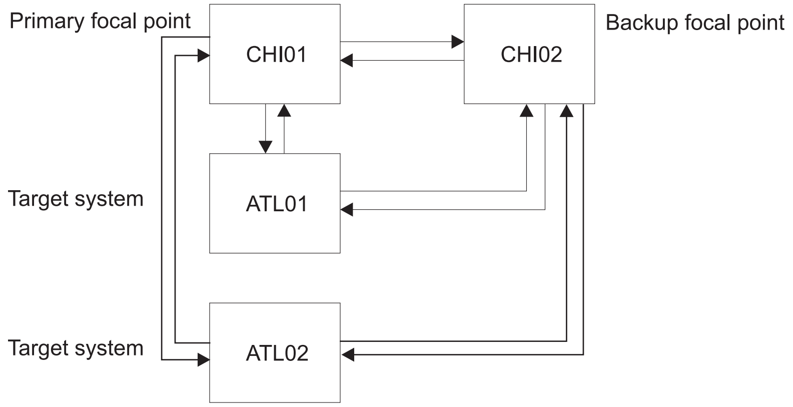
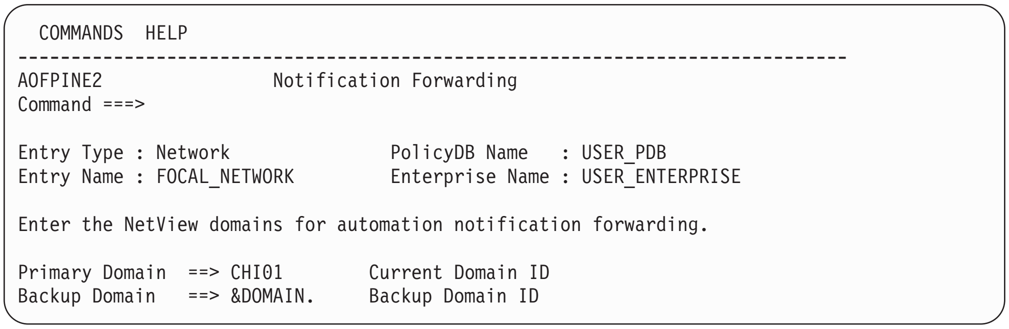


Figure 271. Focal Point Forwarding Definitions for Systems

The corresponding focal-point forwarding definitions in the automation policy for systems CHI02, ATL01 and ATL02 are as follows. You reach the required panel by selecting the FORWARD policy item of a Network entry type.

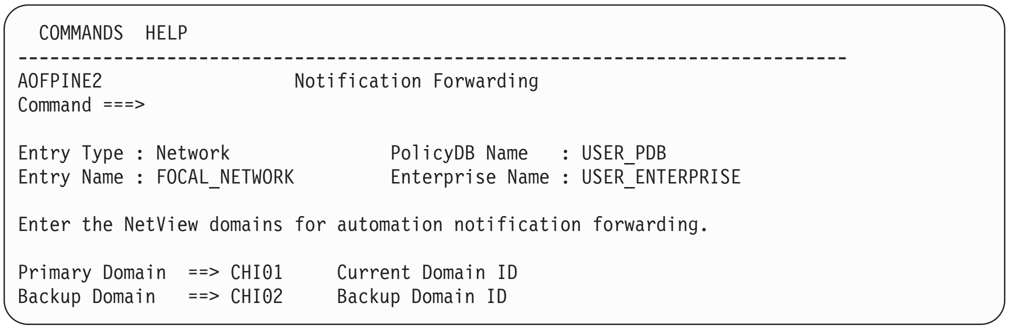
- CHI02 automation policy:



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPINE2                Notification Forwarding
Command ===>

Entry Type : Network                PolicyDB Name   : USER_PDB
Entry Name : FOCAL_NETWORK          Enterprise Name : USER_ENTERPRISE

Enter the NetView domains for automation notification forwarding.

Primary Domain  ==> CHI01         Current Domain ID
Backup Domain   ==> &DOMAIN.      Backup Domain ID
```

Figure 272. Notification Forwarding Panel for CHI02

- ATL01 and ATL02 automation policy:



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPINE2                  Notification Forwarding
Command ===>

Entry Type : Network                PolicyDB Name   : USER_PDB
Entry Name : FOCAL_NETWORK          Enterprise Name : USER_ENTERPRISE

Enter the NetView domains for automation notification forwarding.

Primary Domain  ==> CHI01      Current Domain ID
Backup Domain   ==> CHI02      Backup Domain ID
```

Figure 273. Notification Forwarding Panel for ATL01 and ATL02

In Figure 271 on page 376, the automation policies for system CHI01 do not have any forward focal point definitions. This is because SA z/OS considers the current system as the primary focal point and displays messages *without* forwarding them if either or both of the following is true:

- The forward focal point is not defined in the automation policy for the system.
- The system specified in a forward focal point definition is the current system.

Defining Gateway Sessions

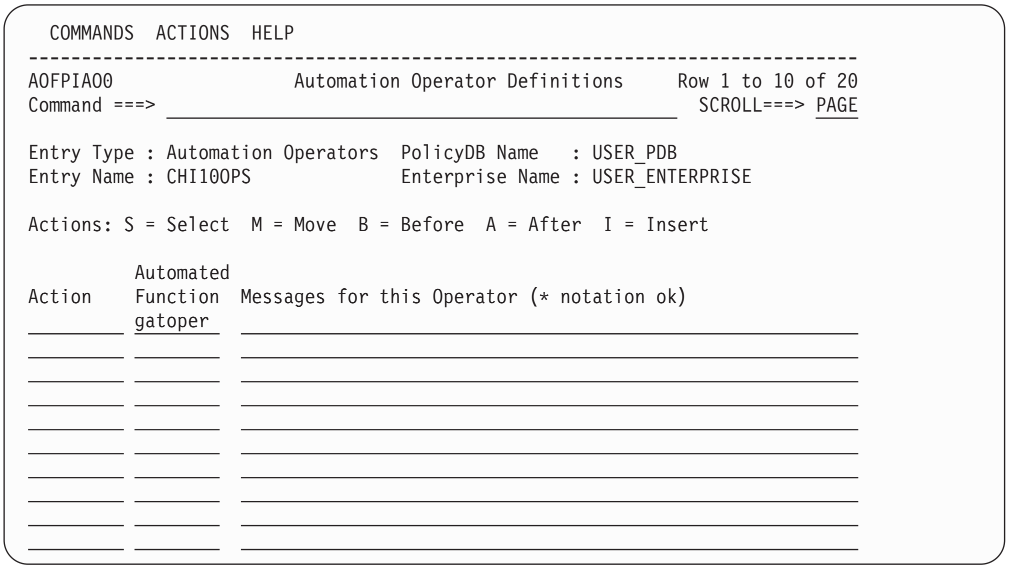
To define gateway sessions:

1. For each system, define the *outbound* gateway autotask (GATOPER) on the Automation Operators policy object of the customization dialog. See "Defining an Outbound Gateway Autotask" for details.
2. On the SA z/OS Network policy object, use the GATEWAY policy item to define the destination systems that the originating system connects to.
3. Define operator IDs used for all inbound and outbound gateway autotasks used on the system in the NetView DSIPARM data set member DSIOPF. See the chapter on how to install SA z/OS on host systems in *IBM Tivoli System Automation for z/OS Planning and Installation* for details.

Defining an Outbound Gateway Autotask

In any system, only the outbound gateway task is defined using the Automation Operators entry type.

If GATOPER has not previously been defined, type the automation operator name, `gatoper`, in the *Automated Function* field of the *Auto Operator Definition* policy item, as shown in Figure 274.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFPIAO0                 Automation Operator Definitions     Row 1 to 10 of 20
Command ===> ______________________________________________   SCROLL===> PAGE

Entry Type : Automation Operators  PolicyDB Name   : USER_PDB
Entry Name : CHI10OPS              Enterprise Name : USER_ENTERPRISE

Actions: S = Select  M = Move  B = Before  A = After  I = Insert

         Automated
Action   Function  Messages for this Operator (* notation ok)
________ gatoper   _________________________________________________________
________ ________  _________________________________________________________
________ ________  _________________________________________________________
________ ________  _________________________________________________________
________ ________  _________________________________________________________
________ ________  _________________________________________________________
________ ________  _________________________________________________________
________ ________  _________________________________________________________
________ ________  _________________________________________________________
________ ________  _________________________________________________________
```

Figure 274. Automation Operator Definitions Panel

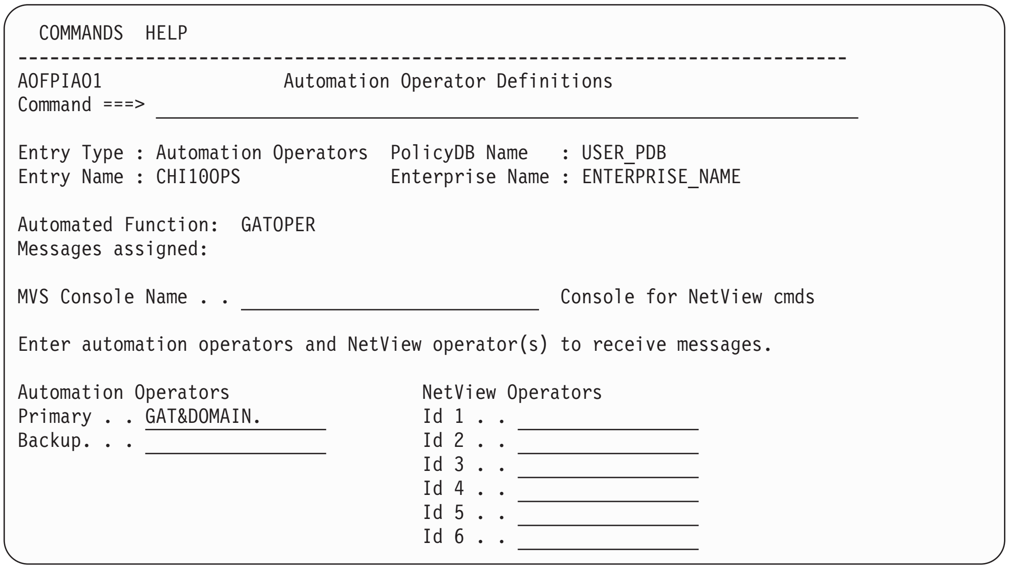
You do not have to specify any messages for GATOPER. When you press Enter, the Automation Operator NetView Userids panel is displayed automatically, as shown in Figure 275 on page 378.

If GATOPER has previously been defined, select it by entering an `S` in the *Action* column.

Enter the NetView operator ID that is associated with the GATOPER function on the Automation Operator NetView Userids panel.

Note:
This NetView operator ID must be unique within the enterprise.

For example, to define the outbound gateway autotask for system CHI01 in the automation network shown in *IBM Tivoli System Automation for z/OS Planning and Installation*, the values shown in Figure 275 are specified.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPIA01                 Automation Operator Definitions
Command ===> __________________________________________________________________

Entry Type : Automation Operators  PolicyDB Name   : USER_PDB
Entry Name : CHI10OPS              Enterprise Name : ENTERPRISE_NAME

Automated Function:  GATOPER
Messages assigned:

MVS Console Name . . ____________________________  Console for NetView cmds

Enter automation operators and NetView operator(s) to receive messages.

Automation Operators                NetView Operators
Primary . . GAT&DOMAIN.______       Id 1 . . _________________
Backup. . . _________________       Id 2 . . _________________
                                    Id 3 . . _________________
                                    Id 4 . . _________________
                                    Id 5 . . _________________
                                    Id 6 . . _________________
```

Figure 275. Automation Operator NetView Userids Panel

Defining Automatically-Initiated TAF Fullscreen Sessions

You can automatically establish Terminal Access Facility (TAF) fullscreen sessions for applications that SA z/OS monitors, so that the operators need not define the sessions on a daily basis.

These TAF fullscreen sessions are defined in the FULL SESSIONS policy item for a Network policy object.

In addition to defining TAF fullscreen sessions using the customization dialog, you follow the NetView process for customizing TAF fullscreen sessions, as outlined in *Tivoli NetView for z/OS Customization Guide*

Once TAF fullscreen sessions are set up, they can be managed using the TAF Fullscreen Menu in the SA z/OS Operator Interface. See *IBM Tivoli System Automation for z/OS User's Guide* for more information on managing TAF fullscreen sessions.

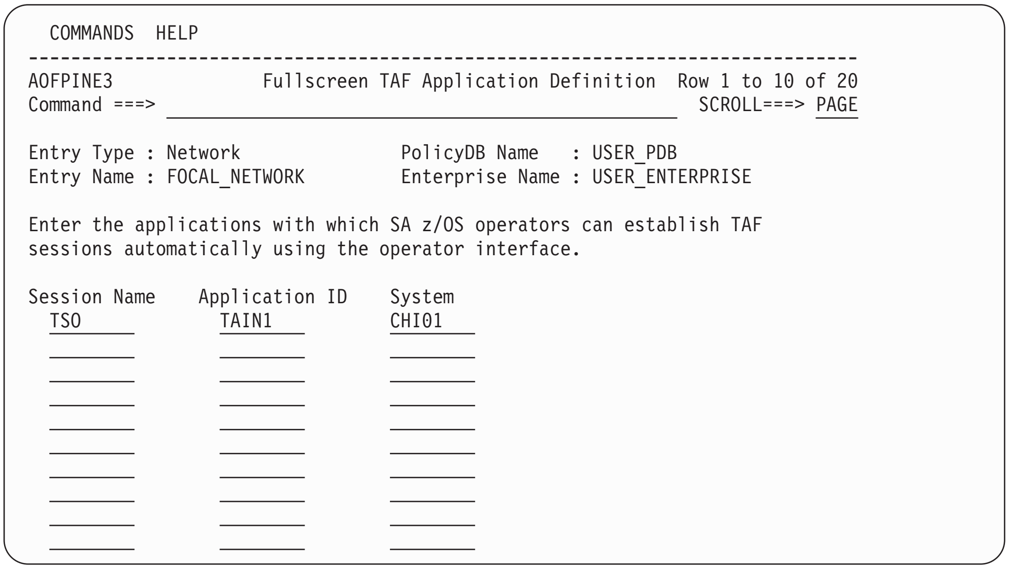
To define an application on the Fullscreen TAF Application Definition panel that you reach by selecting the FULL SESSIONS policy item of a Network policy object, specify the following:

- The session name, or the name of the application that a TAF fullscreen session is to be established for, for example, TSO. This name is displayed in the

Description field on the TAF Fullscreen Menu operator panel. This value can be the same as that used for the application ID.

- The application ID. You can obtain this value from the library containing the network definitions (VTAMLST) or from your network system programmer.
- The system name that the application runs on, for example, CHI01. This is an information-only field and is displayed in the System field on the TAF Fullscreen Menu operator panel.

For example, the following panel defines a TAF fullscreen session for TSO in system CHI01:



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFPINE3             Fullscreen TAF Application Definition  Row 1 to 10 of 20
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Network               PolicyDB Name   : USER_PDB
Entry Name : FOCAL_NETWORK         Enterprise Name : USER_ENTERPRISE

Enter the applications with which SA z/OS operators can establish TAF
sessions automatically using the operator interface.

Session Name    Application ID    System
  TSO_____        TAIN1___        CHI01___
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
  ________        ________        ________
```

Figure 276. Fullscreen TAF Application Definition Panel

Chapter 11. How System Operations Coordinates with Automatic Restart Manager

SA z/OS system operations provides coordination with the Automatic Restart Manager. The Automatic Restart Manager (ARM) is a base z/OS component. It is a recovery function that automatically restarts designated applications when:

- The application abends.
- The system that the application is running on is part of a sysplex, and that system fails. In this case, ARM will attempt to restart the application on another system within the sysplex.

SA z/OS coordinates with ARM to:

- Determine which facility is responsible for restarting a specific application.
- Avoid possible duplications or conflicts in application recovery attempts.
- Allow you to take full advantage of SA z/OS fallback capabilities for applications running on sysplexes. SA z/OS continues to automate an application after it has been moved to a fallback system, provided SA z/OS is installed on that system. If it is not installed on the fallback system, SA z/OS is still aware that the application is active on a system other than its primary one and does not attempt to restart it.

You have to define the Automatic Restart Manager policy using the administrative data utility for ARM policy data (IXCMIAPU) described in *z/OS MVS Setting Up a Sysplex*.

SA z/OS resolves Automatic Restart Manager statuses to SA z/OS statuses, incorporates Automatic Restart Manager-related conditions, and provides one status related to Automatic Restart Manager:

- EXTSTART - The application is being started or restarted externally.

Defining an ARM Element Name

Automatic Restart Manager uses element names to identify the applications with which it works. Each Automatic Restart Manager enabled application must have a unique element name for itself that it uses in all communication with Automatic Restart Manager. Automatic Restart Manager tracks the element name and has its policy defined in terms of element names. If an application moves between systems it MUST continue to use the same element name as it did on the original system. For more information on defining Automatic Restart Manager names to SA z/OS, see “Application Policy Object” on page 156.

All Automatic Restart Manager elements are unregistered initially. Transitions between statuses are caused by:

- IXCARM macro invocations
- Application failures
- System failures
- Timeouts

A minor resource definition *subsystem.0ARM* can be used to tailor automation behaviour during ARM restart processing. As an example, a *subsystem.0ARM* minor resource could be specified with a RESTART EXIT enabled to drive a user supplied exit during ARM restart. The user exit would control additional actions to be taken during ARM restart of the subsystem. If the RESTART flag for this minor resource is resolved to 'N', SA z/OS will not allow ARM to attempt a restart of the application.

Rather than use the *subsystem.0ARM* minor resource definition, a RESTART EXIT could also be specified against the major resource definition for the application. In this case the exit would be driven for all application restarts, not just ARM.

Defining a MOVE Group for Automatic Restart Manager

All resources with the same ARM element name should be linked to one Sysplex Application Group of nature MOVE (MOVE group).

An application's ARM element name is defined either during creation on the Define New Entry panel for applications or after creation via policy item APPLICATION INFO, in both cases in the field *MVS Automatic Restart Management Element Name*.

In order to ensure an application in a MOVE group has completely deregistered from ARM before the automation manager attempts to restart it, a Prepareavailable/WhenObservedDown (passive) relationship must be defined for each ARMed application in the MOVE group with the MOVE group defined as the supporting resource.

To make sure that the automation manager will start the applications linked to a MOVE group, the applications should not be in a HardDown status. The Start On IPL option should not be set to NO.

For more information on how to define MOVE groups refer to "Creating a New ApplicationGroup" on page 134.

Chapter 12. Conversion Function

This chapter provides information on what you should be aware of before using the new release of SA z/OS. In detail, the following topics are discussed:

- "Moved and Restructured Data"
- "Initial Policy Database Conversion" on page 385

The conversion function creates a report called $RPTCNV as a member in the data set of the policy database. All messages are logged.

Moved and Restructured Data

Converting from SA z/OS V3.1

The following changes apply when converting from SA z/OS V3.1:

- Class-instance links are converted to the new data structure needed for the support of nested classes.
- WLM names fields are converted to a new data structure.
- The type specification STANDARD for applications is replaced by specifying `blank` as the type specification. The STANDARD type specification is therefore deleted.
- ASSIST definitions for automation flags are no longer supported. Only the functionality that is formally coded via ASSIST=LOG is still available and must now be coded as AUTO=LOG. Automation flags with ASSIST=LOG or ASSIST=DISPLAY specified are converted to AUTO=LOG if AUTO=YES is found.
- The specification of an SMF ID in the CICS CONTROL policy item is no longer supported due to the new SA z/OS 3.2 SMF recording function.
- Handling of CICS short-on-storage conditions is no longer done via CICS State/Action tables. Therefore any CSA entry type and corresponding links are deleted from the policy database.
- Input parameters for CICS short-on-storage handling are defined via message ID RCVRSOS as in previous SA z/OS releases. To maintain the existing functionality, if the message ID RCVRSOS is defined for an APL entry, it gets a default PASS value of 1 if a PASS or SELECTION value is not available.
- All entries that are found in the MTR entry type are given a default value for the new monitored object specification. The default value that is used is the MTR entry name.
- All data that is defined in the RESOURCE THRESHOLDS policy item for applications of type IMS and CICS is converted to the new MINOR RESOURCE THRES policy item.
- All threshold data that is found in several policy items for entry type MVC is converted to the new MINOR RESOURCE THRES policy item:
  - MVSDUMP thresholds are converted to minor resource thresholds with the resource name MVSESA.MVSDUMP.
  - SMFDUMP thresholds are converted to minor resource thresholds with the resource name MVSESA.SMFDUMP.

  - LOGREC thresholds are converted to minor resource thresholds with the resource name MVSESA.LOGREC, and additionally the resource MVSESA.LOG is created with the same threshold definitions.
  - SYSLOG thresholds are converted to minor resource thresholds with the resource name MVSESA.SYSLOG.
- The JES3 MONITOR policy item for the APL entry type has been deleted. JES3 monitoring can be done via monitor resources (entry type MTR) since SA z/OS 2.3.
- The JES3 ABENDS policy item for the APL entry type has been deleted. Existing error threshold definitions are converted to the new MINOR RESOURCE THRES policy item as definitions for the minor resource name IAT3714. This is the message ID of the specify dump option message at JES3 abend. In contrast to previous SA z/OS releases, threshold settings for IAT3714 are no longer inherited from application and system default settings. To execute the JES3 dump processing function, dependent on the occurrence of the specify dump option message, error thresholds have to be defined for minor resource IAT3714.
- Message ID JESABEND and all definitions that are found for the message ID JESABEND in the MESSAGES/USER DATA policy item for an application are converted to message ID IAT3714.
- The NNT entry type is no longer supported. Any NNT entry types and corresponding links are deleted from the policy database.
- The following data for NTW entry types is no longer supported and is therefore deleted from the policy database:
  - All data from the ADJACENT NETVIEW policy item
  - Logmode and password data for the GATEWAY policy item
- Secondary NMC focal point domains have been removed from the NMC DEFINITIONS policy item in the GRP entry type. You must add existing names manually to the remaining NMC Focal Points field.
- Some field names that are used in the files created via the policy database update function have changed. Thus, a file created with SA z/OS 3.1 may not be compatible with SA z/OS 3.2. It is recommended that you complete all pending policy database updates via file update before migrating to the new release. Otherwise field name changes may be required, or the file might need to be reread from the policy database and updated again.

Restriction:

The subsystem name of application classes must be unique across the whole policy database, that is, it cannot be the same as the subsystem name of any other application instance or class. This is checked for and the conversion does not start if it is not true.

Note also that, depending on the SA z/OS release that you are converting from, a different conversion panel is displayed and different conversion options may apply.

Converting from SA z/OS V2.3

The following changes apply when converting from SA z/OS V2.3:

- Subsystem defaults data have been moved from entry type SDF to entry type ADF.
- SVPs have been deleted from Triggers.
- Entry type ICL has been deleted.

- Entry type DEN has been deleted.
- WTOBUF data has been deleted from entry type MVC.
- ENVIRON RACF data has been deleted from entry type NTW.
- Console ID has been deleted from entry type AOP.
- Console ID has been deleted from ENVIRON CONSOLE in entry type SYS.
- RCVRVIOL and RCVRAUTO message IDs have been deleted from entry type APL.
- The monitor routine defaults to NONE for non-MVS APLs.
- Minor resource thresholds have been deleted from DB2 APLs.
- Minor resource thresholds have been deleted from CICS APLs for resource AUTO and VIOL.

Initial Policy Database Conversion

The first time that a policy database is opened with the customization dialog, conversion to the current layout is done automatically. There are special conversion options available by selecting **0 Settings** in the Customization Dialog Primary Menu, and then **4 Convert** in the Customization Dialog Settings Menu. This displays the PDB Convert Option panel. This panel is used to specify the options for the PDB Conversion. These options are saved in your ISPF profile pool and remain in effect until set again

On this panel, SA z/OS allows you to request the generation of an automation name for application groups during the very first conversion of a policy database. If an application group has an automation name, SA z/OS can generate resources from such entries.

Selective Policy Database Conversion

After your entries have been migrated to the policy database, the conversion is done to convert the old data which might have been migrated to the policy database to the new format. The migration functions automatically start the conversion function.

Generating a Job for Batch Conversion

You can run a job for batch conversion that performs the same processing as the conversion function of the customization dialog. The SINGSAMP sample library contains a sample job for this task in member INGEBCNV. Before submitting this job, you need to edit the jobcard appropriately as described in the sample itself.

Appendix A. Concurrent Customization by Multiple Users

Concurrent user access is supported for entry types SYS, APG, APL, MTR, and UET. These are referred to as "multi-user capable entry types".

The following subsections describe the consequences for users that want to work simultaneously on entry types within one policy database.

Concurrent User Access Capabilities

If a policy database is selected for processing (using the OPEN command), access to that policy database for other users is possible, that is, multiple opening of a policy database is supported.

Working on entry types other than the multi-user capable ones with several users, or performing commands other than OPEN or SELECT against a policy database while an entry in a multi-user capable entry type is being edited, may not work.

This limitation depends on the data tables that are needed for the selected action or command, and it is indicated by specific messages. The most common situations are described in the following sections.

Policy Definitions Usage

If you want to change your Enterprise Policy Options by selecting option 10 on the Customization Dialog Settings Menu panel (see Figure 277 on page 392), while another user works with a policy database, the selection is not possible and an appropriate message is displayed.

On the other hand, if you are accessing the SA z/OS policy items for modification, this will prevent other users from accessing any policy item and an appropriate message is displayed.

Policy Database Selection

If you want to issue a BUILD, DELETE, or REPORT command from the Policy Database Selection panel or use a DATA MANAGEMENT function while another user has accessed the same policy database, the command is rejected and an appropriate message is displayed.

Conversely, if you request access to a policy database via *S (Select)* or *O (Open)*, while a BUILD, DELETE, DATA MANAGEMENT function, or REPORT is in progress for this policy database, the access is rejected and an appropriate message is displayed.

For other commands there are also limitations, if a user has already opened the specific policy database. These limitations depend upon the selected entry types and actions, and are indicated by an appropriate message.

Entry Type Selection

If a policy database is opened successfully (with *S (Select)* or *O (Open)*), the Entry Type Selection panel is displayed. Here a specific Entry Type can be selected, or a BUILD can be requested.

If a BUILD command is issued from this screen, but another user has also opened this policy database to work with applications or any other entry type, the command is rejected and an appropriate message is displayed.

If an entry type other than the multi-user capable ones is selected, and another user has also selected an entry type of the same policy database, the selection may be rejected depending on the data tables needed for the selected entry type, and an appropriate message is displayed. In this case, editing is not possible, because concurrent user access is provided only for the multi-user capable entry types.

Entry Name Selection

On the Entry Name Selection panel a specific application can be selected, and a BUILD can be requested.

If you try a BUILD action, and the policy database has been simultaneously opened by another user, the command is rejected and an appropriate message is displayed.

If another user is currently editing the same application in the same policy database, access is allowed only to view the data, that is, changing the displayed data is not allowed. In this case, however, it is possible to select that application and view the data. A warning message is displayed on the Policy Selection panel, indicating that no updates will be accepted.

If, nevertheless, the displayed values are changed on the data entry panel for that policy option, these changes are not accepted when leaving that panel. If you try to leave the panel with the `END` command, a message will remind you that you must leave this panel with `CANCEL`.

Changing Links

The WHERE USED policy item for Systems allows you to change the links of one system to one or more groups or subgroups.

The WHERE USED policy item for Applications allows you to change the links of one application to one or more application groups.

The WHERE USED policy item for ApplicationGroups allows you to change the links of one application group to one or more systems or sysplexes.

The WHERE USED policy item for Monitor Resources allows you to change the links of one monitor resource to one or more monitor resources.

You can only add or remove such links, if you exclusively access the entry types System, Application, ApplicationGroup, or Monitor Resource. If another user is currently editing them, a message is displayed indicating that updates are not possible.

The same is true for the trigger policy item where you can maintain the link of the current application, application group, or monitor resource to a trigger.

Automated NetView Automation Table Build

If you have the automated AT and MPFLSTSA build enabled (this is the default behavior), each policy change that affects the ATs or MPFLSTSA member is recorded for each system. This requires the exclusive access of entry type SYSTEM. If another user is currently using these files, a message is displayed that updates are not possible.

Appendix B. More about Using the Customization Dialog

Use this appendix to find out how to get help when using the customization dialog and how to change the customization dialog environment.

Getting Help

Online help is available for the customization dialog. You can display online help by entering the HELP command (or press the PF key associated with the HELP command) while on a particular dialog panel.

When selecting HELP from the action bar, the pull-down lists the following selections:

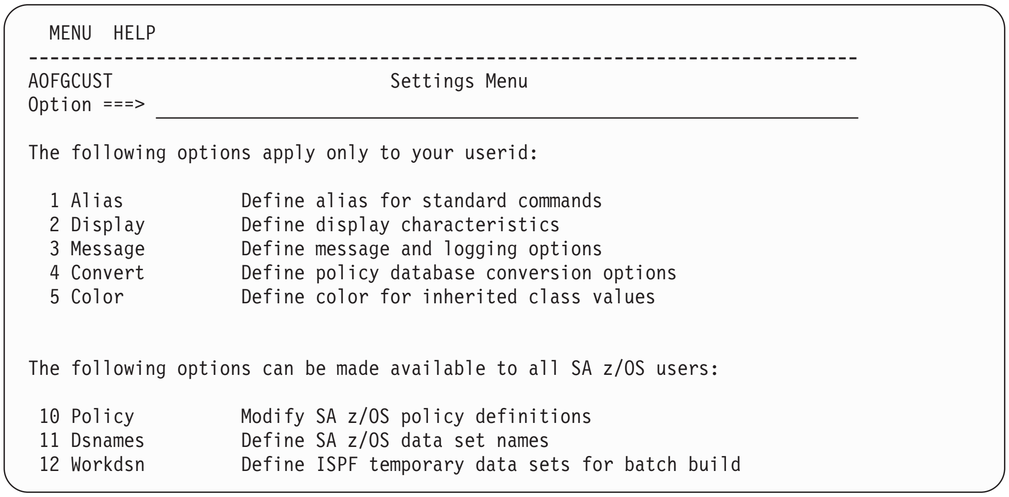
Automation Help
This help provides online information about important functions of the customization dialog. It combines the help panels of the functional panels within the customization dialog. This combination allows you to view the help panels sequentially, or you can choose selected topics from lists that are displayed on many of the help pages.

Changes for this Release
This panel provides a list of the changes and enhancements that have been added to the customization dialog for the different releases of version 2.

Changing Your Customization Dialog Environment

You can invoke the Settings function from the Customization Dialog Primary Menu or from the Menu pull-down of the Policy Database Selection panel.

The first three options in the Customization Dialog Settings Menu panel (see Figure 277 on page 392) allow you to change the customization dialog environment for your user ID. This is a usability function only and has no effect on the policy database contents.



```
  MENU  HELP
------------------------------------------------------------------------------
AOFGCUST                          Settings Menu
Option ===>

The following options apply only to your userid:

  1 Alias           Define alias for standard commands
  2 Display         Define display characteristics
  3 Message         Define message and logging options
  4 Convert         Define policy database conversion options
  5 Color           Define color for inherited class values


The following options can be made available to all SA z/OS users:

 10 Policy          Modify SA z/OS policy definitions
 11 Dsnames         Define SA z/OS data set names
 12 Workdsn         Define ISPF temporary data sets for batch build
```

Figure 277. SA z/OS Customization Dialog Settings Menu

Alias This option allows you to change commands and command abbreviations that can be entered on the command line or action column in the customization dialog.

Display
This option allows you to change the line spacing for lists that appear in the customization dialog.

Message
This option allows you to specify the level of detail in the messages you receive when you run various customization dialog options or batch jobs, and whether the messages are written to the ISPF log.

These settings are stored in the ISPF application profile. To have them available for batch processing (build or migration), make sure that your ISPF profile data set is allocated to the table library, DDname: AOFTLIB, as described in “Defining Data Sets for Batch Processing” on page 396.

Convert
This option lets you specify whether SA z/OS should create an automation name for an ApplicationGroup resource during the very first conversion of a policy database. Refer to “Initial Policy Database Conversion” on page 385 for more information.

Color This option allows you to specify the color and reverse video option that is used to display inherited class values.

Policy See “Extending Policy Definitions” on page 393;

Dsnames
See “Defining Data Sets for Batch Processing” on page 396;

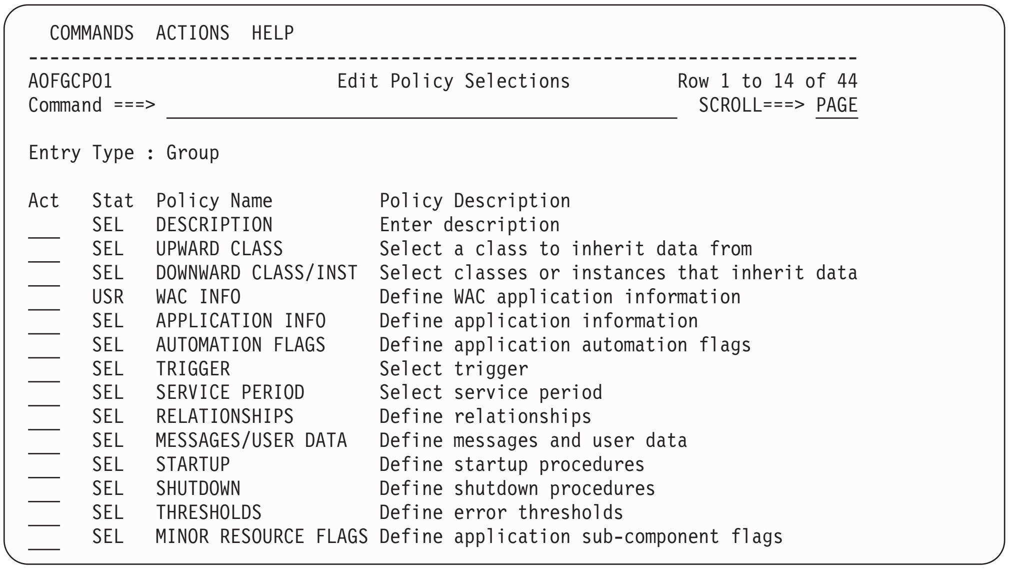
Workdsn
See “Define ISPF Temporary Data Sets for Batch Build” on page 398;

Extending Policy Definitions

With SA z/OS you can extend your policy definitions by creating new policy items or modifying existing ones. Any user-defined policy items that you create or modify are now kept separate from the SA z/OS policies. This means that with future releases of SA z/OS you will not have to redefine those policy items that you have created or modified.

To create new policy items or modify existing ones, you should perform the following steps:

1. Select option **10 Policy** from the Customization Dialog Settings Menu (see Figure 277 on page 392). An Entry Type Selection panel is displayed.
2. Select the entry type that you want to create a new policy item or modify one for. The Edit Policy Selections panel is displayed, listing all current policy items for that entry type, as shown in Figure 278. Any user-defined policy items are indicated by USR in the **Status** column.



```
  COMMANDS  ACTIONS  HELP
------------------------------------------------------------------------------
AOFGCPO1                     Edit Policy Selections         Row 1 to 14 of 44
Command ===> ________________________________________________  SCROLL===> PAGE

Entry Type : Group

Act   Stat  Policy Name          Policy Description
___   SEL   DESCRIPTION          Enter description
___   SEL   UPWARD CLASS         Select a class to inherit data from
___   SEL   DOWNWARD CLASS/INST  Select classes or instances that inherit data
___   USR   WAC INFO             Define WAC application information
___   SEL   APPLICATION INFO     Define application information
___   SEL   AUTOMATION FLAGS     Define application automation flags
___   SEL   TRIGGER              Select trigger
___   SEL   SERVICE PERIOD       Select service period
___   SEL   RELATIONSHIPS        Define relationships
___   SEL   MESSAGES/USER DATA   Define messages and user data
___   SEL   STARTUP              Define startup procedures
___   SEL   SHUTDOWN             Define shutdown procedures
___   SEL   THRESHOLDS           Define error thresholds
___   SEL   MINOR RESOURCE FLAGS Define application sub-component flags
```

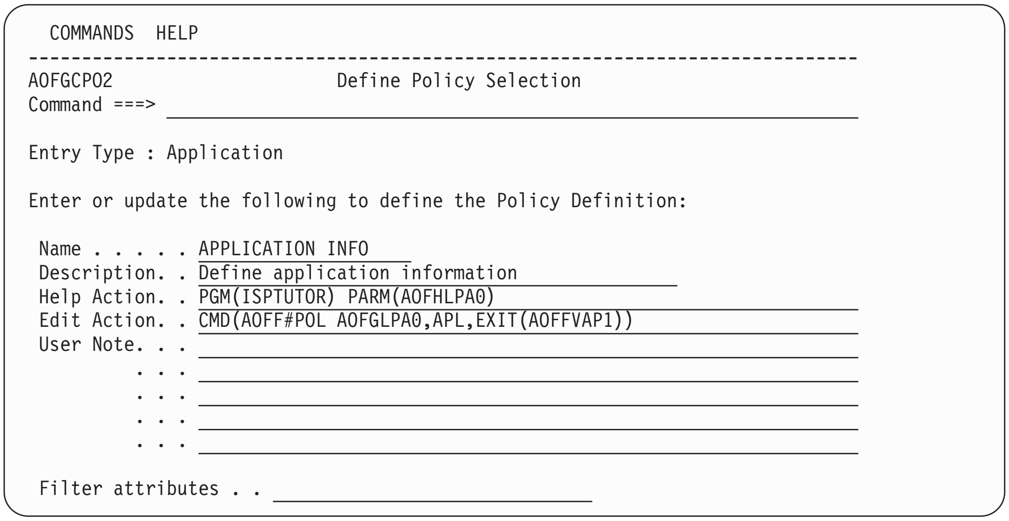
Figure 278. Edit Policy Selections Panel

3. In the Edit Policy Selections panel you can:
   - Create a new entry by entering `NEW` at the command line. The new entry is added to the top of the list.
   - Create a new entry by entering `INSERT` in the **Act** column. The new entry is added after the entry where INSERT was specified.
   - Edit an entry by entering EDIT in the **Act** column.

     Note:
     Any modifications to the existing SA z/OS policy definitions are solely your responsibility.

   - Delete a user-defined entry by entering DELETE in the **Act** column.
   - Suppress an SA z/OS entry by entering REMOVE in the **Act** column.
   - Activate a deselected entry by entering SELECT in the **Act** column.
   - Leave the panel and save all changes by entering the END command.
   - Discard all changes by entering the CANCEL command.

4. If you choose to insert a new policy item for an entry type, or edit an existing policy item, the Define Policy Selection panel is displayed. The example in Figure 279 shows how to define policy item APPLICATION INFO of entry type Application



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGCPO2                     Define Policy Selection
Command ===>

Entry Type : Application

Enter or update the following to define the Policy Definition:

 Name . . . . . APPLICATION INFO
 Description. . Define application information
 Help Action. . PGM(ISPTUTOR) PARM(AOFHLPA0)
 Edit Action. . CMD(AOFF#POL AOFGLPA0,APL,EXIT(AOFFVAP1))
 User Note. . .
          . . .
          . . .
          . . .
          . . .

 Filter attributes . .
```

Figure 279. Define Policy Selection Panel

If you want to create policy items you must specify an ISPF action for the selection of the new policy item from the Policy Selection panel. You must then create your own ISPF input panel for the new policy item.

Each policy selection consists of information defining:

- How the policy appears on the *Policy Selection* panel.
- How the policy is processed when it is selected.
- Under what conditions the policy should be displayed.

This panel has the following fields:

Entry Type
Indicates the entry that the policy is being defined for. This field is set based on the entry type you had previously selected.

Policy Name
This is the name of the policy you are defining. It appears in the first column of the Policy Selection panel. This field is required.

Description
This is a short description of the policy. It appears in the second column of the Policy Selection panel. This field is required.

Help Action
Specifies the action to be taken when policy help is requested (by entering HELP in the Action column on the Policy Selection panel). The action must be a valid ISPF SELECT service. For example, to display an ISPF panel enter:

```
PGM(ISPTUTOR) PARM(panel-name)
```

The help text should describe the function provided by this policy rather than describing how to enter the data for the policy.

This field is optional; if no action is specified HELP will not be available from the Policy Selection panel for this policy.

Edit Action
Specifies the action to be taken when the policy is selected from the Policy Selection panel. The action must be a valid ISPF SELECT Service. This field is required.

User Note
These fields are not shown on the Policy Selection panel. They can be used for documentation purposes. These fields are optional.

You can also specify filtering conditions, that is, attributes of the entry being processed from the Policy Selection panel that can be used to determine whether this policy should be displayed. The following filtering information can be specified:

Filter attributes
This specifies attributes that an entry must have in order for this policy to be displayed. This field contains multiple words, each word identifying one attribute. All the attributes specified must be satisfied in order for the policy to be displayed. If no attributes are specified then the policy is valid for all entries. The attributes that can be specified are defined by SA z/OS and differ based on the entry type, as shown in Table 10.

Table 10. Policy Selection Filter Attributes

Entry Type	Attribute	Value
Group	Group type	(blank) STANDARD SYSPLEX
System	Operating System type	(blank) CF LINUX MVS VM VSE
ApplicationGroup	ApplicationGroup type	(blank) SYSPLEX SYSTEM
Application	Object type	(blank) CLASS INSTANCE
	Application type	(blank) IMAGE JES2 JES3 CICS IMS DB2 OPC USS
Processor	Processor type/mode	(blank) LPAR ESA PSM
Others	–	

The following rules apply to the filtering attributes:

1. If no attributes are specified (indicated by *(blank)* in Table 10 on page 395), the policy is always displayed for all entries.
2. If multiple attributes are specified then all attributes must be satisfied in order for the policy to be displayed. For example:

   `JES2 CLASS` This displays the policy for application entries that are classes and are of type JES2.

   `JES2` This displays the policy for application entries that are of type JES2 and either class or instances.
3. For each ATTRIBUTE, the words listed in the VALUE column are mutually exclusive. If you specify both values the policy will never be displayed as both conditions cannot be true at the same time. For example:

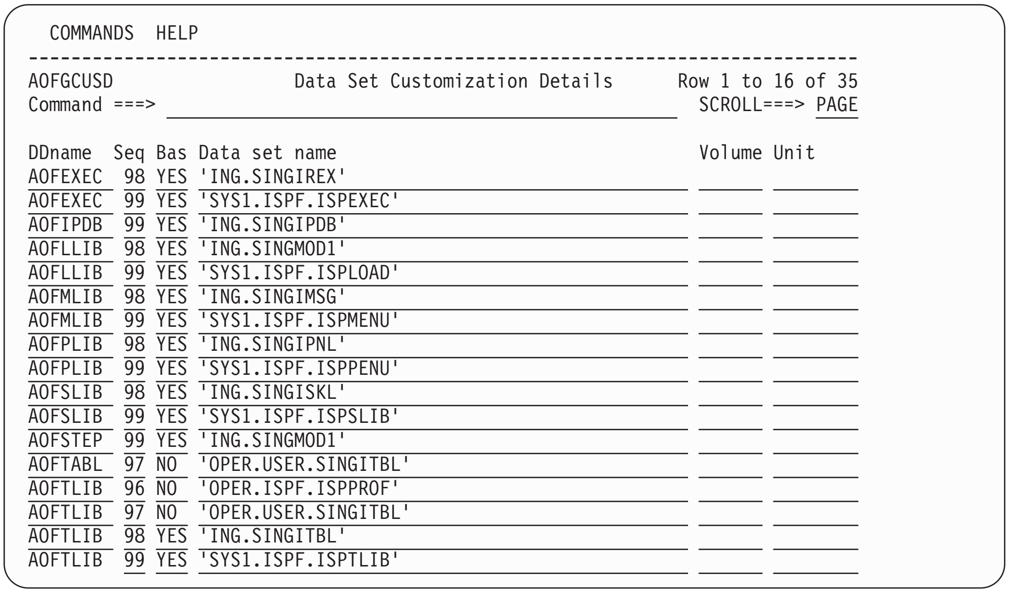
   `JES2 JES3` This is never displayed because an application can be either JES2 or JES3 but not both.
4. No validity checking is done on the attribute values. If an invalid value is specified the policy will never be displayed since the condition cannot be satisfied. For example:

   `JES4` This is never displayed because an application cannot be of type JES4.

Defining Data Sets for Batch Processing

Use Option 11 of the Customization Dialog Settings Menu panel to define the batch processing data sets. This needs to be done only once when you install SA z/OS. For details, see *IBM Tivoli System Automation for z/OS Planning and Installation*.

To use the batch options of the build or migrate function you must first specify the names of the data sets required for the batch processing. These names form the DD name concatenations to be included in the JCL of the batch processing job.



```
  COMMANDS  HELP
------------------------------------------------------------------------------
AOFGCUSD                 Data Set Customization Details      Row 1 to 16 of 35
Command ===>                                                  SCROLL===> PAGE

DDname  Seq Bas Data set name                                 Volume Unit
AOFEXEC  98 YES 'ING.SINGIREX'
AOFEXEC  99 YES 'SYS1.ISPF.ISPEXEC'
AOFIPDB  99 YES 'ING.SINGIPDB'
AOFLLIB  98 YES 'ING.SINGMOD1'
AOFLLIB  99 YES 'SYS1.ISPF.ISPLOAD'
AOFMLIB  98 YES 'ING.SINGIMSG'
AOFMLIB  99 YES 'SYS1.ISPF.ISPMENU'
AOFPLIB  98 YES 'ING.SINGIPNL'
AOFPLIB  99 YES 'SYS1.ISPF.ISPPENU'
AOFSLIB  98 YES 'ING.SINGISKL'
AOFSLIB  99 YES 'SYS1.ISPF.ISPSLIB'
AOFSTEP  99 YES 'ING.SINGMOD1'
AOFTABL  97 NO  'OPER.USER.SINGITBL'
AOFTLIB  96 NO  'OPER.ISPF.ISPPROF'
AOFTLIB  97 NO  'OPER.USER.SINGITBL'
AOFTLIB  98 YES 'ING.SINGITBL'
AOFTLIB  99 YES 'SYS1.ISPF.ISPTLIB'
```

Figure 280. SA z/OS Data Set Customization Details Panel

On the Data Set Customization Details panel, a table is displayed with groups of data definition (DD) names and data set names. You must modify the data set names in accordance with your local naming conventions.

Each column can be modified as follows:

DDname
No new DD names can be added, but a new entry for an existing DD name can be added at the end of the table. It is inserted at the correct position within the table when it is saved. An entry can be deleted by blanking out the DD name, but if an entry is removed incorrectly the batch functions become inoperative.

Seq This entry should be a number from 1 through 99. This value positions the entry in the correct concatenation sequence when the JCL is built. The higher the number, the further down the concatenation sequence it appears. The values 98 and 99 are reserved.

Base The Base column indicates a reserved DD name entry and should not be removed. All entries that are not reserved (see the **Seq** column description) must have the value of NO.

Data set name
This is the fully-qualified data set name, including quotation marks, that is represented by this DD name.

Volume
May be left blank for catalogued data sets, otherwise enter the label ID of the volume where the data set is located (as would be coded for an ALLOC JCL entry). SMS may be entered for any data set under space management.

Unit Must be left blank for catalogued and space managed data sets, otherwise you must enter the unit relevant to the volume where the data set is located (as would be coded for an ALLOC JCL entry).

AOFIPDB is needed to get access to the automation manager logic file.

For AOFTLIB the following data sets may be additionally allocated to the ISPF and SA z/OS table libraries:

- The user’s ISPF table output library is needed to get access to the tables with user modifications, for example, the data set specifications
- The user’s ISPF profile may be added to obtain access to the dialog, messages and logging options. If you omit this, the options that provide the most information are always used. This relates only to the settings in option 3 Message of the Customization Dialog Settings Menu panel.

In the example in Figure 280 on page 396, these are:

```
'OPER.ISPF.ISPPROF'
'OPER.USER.SINGITBL'
```

To avoid ISPF enqueues with TSO users a temporary data set is always added as the first data set of the ISPTLIB data set allocations.

Note:

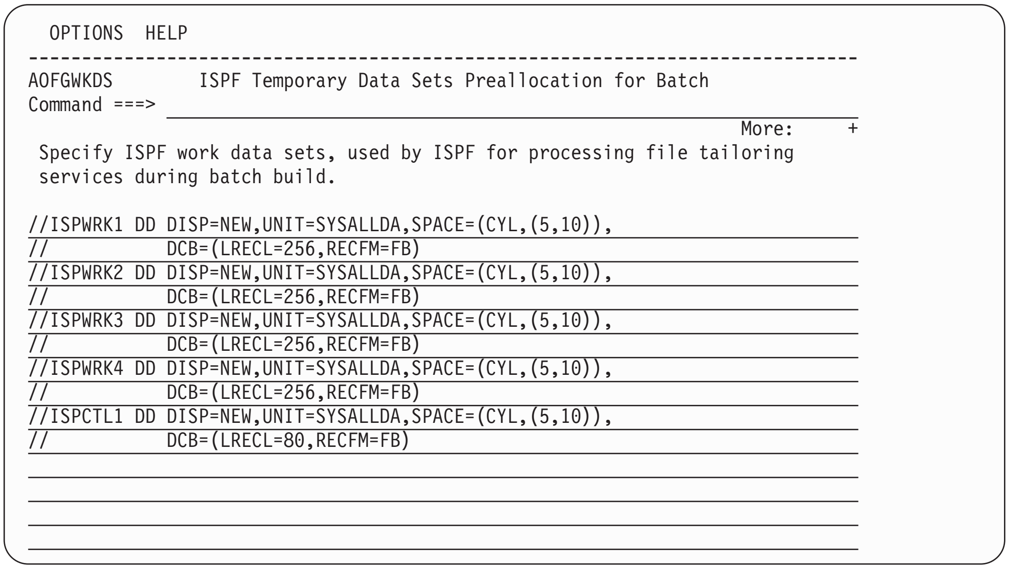
The user's ISPF table output library should also be allocated to DDname AOFTABL for functions such as the PDB report or updating the PDB using text files.

The ISPF load libraries should be allocated to DDname AOFSTEP if they have not been defined in LPA.

To avoid ISPF enqueues with the ISPF input table library name, a temporary data set is always added as the first data set of the ISPTLIB data set allocation.

Define ISPF Temporary Data Sets for Batch Build

Use Option 12 in the Customization Dialog Settings Menu panel to define the data sets for the batch build. SA z/OS displays the ISPF Temporary Data Sets Preallocation for Batch panel (see Figure 281).



```
  OPTIONS  HELP
------------------------------------------------------------------------------
AOFGWKDS        ISPF Temporary Data Sets Preallocation for Batch
Command ===> 
                                                             More:     +
 Specify ISPF work data sets, used by ISPF for processing file tailoring
 services during batch build.

//ISPWRK1 DD DISP=NEW,UNIT=SYSALLDA,SPACE=(CYL,(5,10)),
//           DCB=(LRECL=256,RECFM=FB)
//ISPWRK2 DD DISP=NEW,UNIT=SYSALLDA,SPACE=(CYL,(5,10)),
//           DCB=(LRECL=256,RECFM=FB)
//ISPWRK3 DD DISP=NEW,UNIT=SYSALLDA,SPACE=(CYL,(5,10)),
//           DCB=(LRECL=256,RECFM=FB)
//ISPWRK4 DD DISP=NEW,UNIT=SYSALLDA,SPACE=(CYL,(5,10)),
//           DCB=(LRECL=256,RECFM=FB)
//ISPCTL1 DD DISP=NEW,UNIT=SYSALLDA,SPACE=(CYL,(5,10)),
//           DCB=(LRECL=80,RECFM=FB)
```

Figure 281. ISPF Temporary Data Sets Preallocation for Batch Panel

This panel allows you to add or modify job control data to preallocate temporary ISPF data sets that are used for the batch build. The build function creates the ACF and AMC output files in two steps:

1. All members are created and stored in a temporary file using ISPF file tailoring.
2. This file is written to disk creating all the individual members.

To hold the intermediate files, ISPF work and control data sets are used. If the default allocations of ISPF are not sufficient, they can be changed on this panel.

The ISPWRK*x* ddnames are used by ISPF for file tailoring services. The ISPCTL*x* ddnames are used by ISPF to hold the file tailoring output and to pass it to the subsequent functions that write the members to the output data set.

User-written Functions

Option **U User** on the Customization Dialog Primary Menu displays the User-written Functions Selection Menu. This menu allows you to add installation-specific functions to the customization dialog.

As an example, one selection has already been provided:

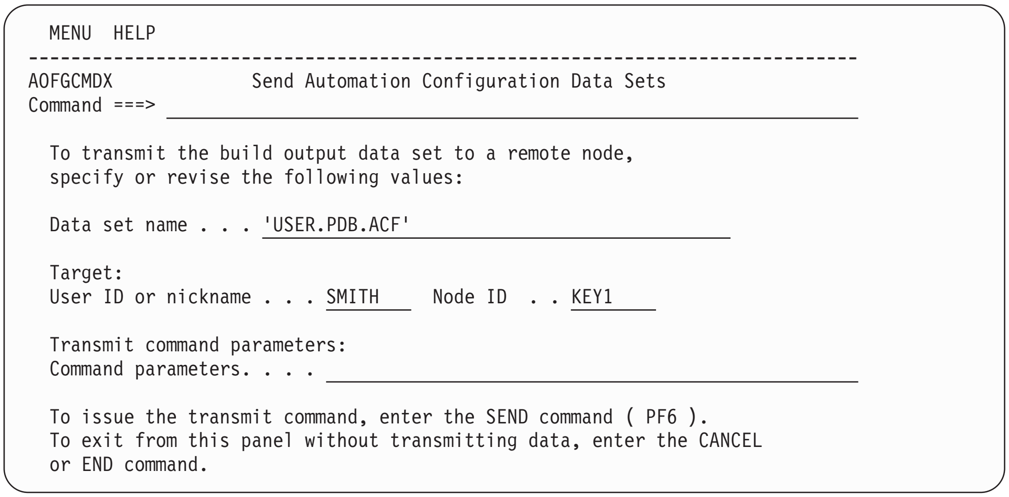
```
1 - Transmit      Transmit build output to a remote node
```

If you select this option 1, SA z/OS displays the Send Policy Data Base panel (see Figure 282), where you can distribute policy databases to remote nodes.

To add further selections, edit the panel (member AOFGUSR) and do the following:

- Add selection number and description to the panel)BODY section
- Add a line to the TRANSLATE instruction in the panel)PROC section, containing the selection number and the command string which invokes the new function.

The Send Policy Data Base panel allows you to transmit a build output data set to a specified node and user ID.



```
  MENU  HELP
------------------------------------------------------------------------------
AOFGCMDX            Send Automation Configuration Data Sets
Command ===> _________________________________________________________________

  To transmit the build output data set to a remote node,
  specify or revise the following values:

  Data set name . . . 'USER.PDB.ACF'____________________________

  Target:
  User ID or nickname . . . SMITH___   Node ID  . . KEY1____

  Transmit command parameters:
  Command parameters. . . . __________________________________________________

  To issue the transmit command, enter the SEND command ( PF6 ).
  To exit from this panel without transmitting data, enter the CANCEL
  or END command.
```

Figure 282. Send Automation Configuration Data Sets Panel

With the parameter specified on this screen, a TSO TRANSMIT command is created. The transmission is started by issuing the SEND command (PF 6).

Data set name
: This entry field is filled with the build output data set name of the current policy database, if that data set name is available. The name is always fully qualified and follows the TSO data set naming conventions.

User ID or nickname
: allows to specify the target user ID or nickname. Input in this field is saved across sessions.

Node ID
: allows to specify the target node. This field may remain empty, if a nickname has been specified. Input in this field is saved across sessions.

Command parameters
allows to specify parameters for the TRANSMIT command. Any values specified here are appended to the TRANSMIT command as they have been typed in. Input in this field is saved across sessions.

Alternate Configuration Support

There may be situations, for example, in disaster recovery or on weekends, when you want to have an alternative automation configuration for one or more systems out of your enterprise (policy database).

The customization dialog allows you to specify an alternate configuration by:

- marking a system as part of an alternate configuration
- allowing a specific build for normal or alternate configurations
- enforcing the build of normal or alternate configurations into different output data sets

All other definitions and links remain unchanged, for example, an application group can be linked to a normal systems, an alternate systems, or to both.

How to Define Alternate Configurations for Systems

For each system, either on the Define New Entry panel or in the policy item SYSTEM INFO, the existing MVS system name in field *MVS SYSNAME* allows you to add a ″+″ sign to the system name. This will be used as an indication that this system belongs to the alternate configuration. The ″+″ sign can only be added to the end of the system name.

Furthermore, this ″+″ sign can be specified in the supporting resource definition of a relationship, and on the initial migration panel.

When a new system is defined and its processor operations name (in field *ProcOps name*) is defaulted with the MVS system name, any ″+″ is removed.

When linking systems to a group or subgroup, the configuration type is not checked. If for example, a sysplex is intended for an alternate configuration, you have to ensure that only systems for this configuration are linked to that sysplex.

How to Build Alternate Configurations

The Build Parameters panel (see Figure 258 on page 340) has an option to select either a normal or alternate configuration build. The default is NORMAL. To avoid overwriting an existing configuration with wrong data, the build checks the configuration type in the build output data set. If there is already a configuration of a different type, the build is not started, and a warning message is displayed. If the policy database contains a normal and an alternate configuration, they have to be built into different output data sets.

Furthermore, the policy database name is checked to prevent an automation control file being destroyed by building a configuration into the wrong output data set.

When building an alternate configuration, all ″+″ signs are removed from the MVS system name. Thus the generated automation configuration files can be used for automation simply by switching the data sets.

If a policy database contains two configurations, the build picks out the systems that are linked to the selected configuration:

- The ACF map AOFACFMP contains only the systems marked for the selected configuration via the ″+″ sign added to the MVS system name.
- System include members (ACFZ*xxx*) are generated only for the systems of the selected configuration.
- The AMC file, which is sysplex oriented, consists of members for each sysplex and standalone system. In each sysplex, it contains only the systems that are marked for the selected configuration, and standalone system members are created only for systems of the selected configuration.
- The ATs are generated only for the systems of the selected configuration.
- The MPFLSTSA member is built for the messages that belong to the selected configuration.

Restrictions

For performance reasons, building the complete policy database (option 1), does not check which system each object is linked to. Thus the build output data set will contain fragments of each object in the policy database, regardless of which configuration they are linked to. Of course, fragments of the wrong configuration cannot be accessed by automation because they are not included via system main include members. However, the build output data sets need to be large enough to hold all fragments.

Furthermore, with build option 3 (to build an entry type or entry name), it is possible to update each fragment in a data set, independent of its link to a configuration.

The *Unlinked* report (see Chapter 7, “Creating Reports about Policy Databases,” on page 329) does not include relationships that have been defined across configurations. That means, if a relationship is defined between a resource in configuration NRM and another resource in configuration ALT, it is not included by the build function. However, the *Unlinked* report does not show it as not used.

It is not possible to define processor operations proxy resources for systems belonging to the alternate configuration, because the Processor Operations name must be different for all systems. For examples on how to use proxy definitions see *IBM Tivoli System Automation for z/OS Customizing and Programming*.

Appendix C. Example New Policy Objects via File Updates

As discussed in “Making Updates with Text Files” on page 356, it is possible to make bulk updates of policy objects using a flat text file. You can also edit the text file to add new policy objects (for more details see “Creating New Policy Objects via File Updates” on page 362).

To create new policy objects, insert sections that have the following structure:

```
NEW <Entry Type>                  : <new entry name>
COPY                              : <name of existing entry to copy data from>
WHERE_USED <parent entry type> : <name of existing entry to link to>
LINK_TO_CLASS                     : <name of existing APL class entry to link to>
```

You can create new policy objects for any entry type *except* for type ENT (enterprise). All the fields that are available on the NEW panel for the entry type that you are creating are also valid here.

This appendix provides examples for the following entry types:

- APL
- APG
- MTR
- UET

These are the most common entry types that you are likely to want to create new policy objects for via file updates.

Application

The following is an example for a new IMS Application Class entry, APLNEW, with data copied from an existing application, OTHERAPL, and linked to another class, IMSCLASS1:

```
NEW APL                : APLNEW
  Subsystem Name       : APLNEW
  Object Type          : CLASS
  Application Type     : IMS
  Subtype              : CTL
  Job Type             :
  Job Name             :
  Transient Rerun      :
  Scheduling Subsystem :
  JCL Procedure Name   :
  Short Description    : New IMS application
  Long Description 1   :
  Long Description 2   :
  Long Description 3   :
  Long Description 4   :
  Long Description 5   :
COPY                   : OTHERAPL
LINK_TO_CLASS          : IMSCLASS1
```

Application Group

The following is an example for a new system Application Group entry, APGNEW, with data copied from an existing application group, OTHERAPG, and linked to an existing system, SYSTEM1:

```
NEW APG               : APGNEW
  Type                : SYSTEM
  Nature              : BASIC
  Default Preference  : *DEF
  Automation Name     : APGAUTO
  Automatically link  : YES
  Behaviour           : ACTIVE
  Move Mode           :
  Short Description   : New Application Group  of type System
  Long Description 1  :
  Long Description 2  :
  Long Description 3  :
  Long Description 4  :
  Long Description 5  :
COPY                  : OTHERAPG
WHERE_USED SYS        : SYSTEM1
```

Monitor Resource

The following is an example for a new Monitor Resource entry, MTRNEW:

```
NEW MTR               : MTRNEW
  Monitored object    : OBJECT1
  Monitored jobname   : JOB1
  Short Description   : New Monitor Resource
  Long Description 1  :
  Long Description 2  :
  Long Description 3  :
  Long Description 4  :
  Long Description 5  :
```

User Entry-Type Pair

The following is an example for a new User E-T Pair entry, UETNEW:

```
NEW UET               : UETNEW
  Short Description   : New User E-T Pair.
  Long Description 1  : For this entry type only
  Long Description 2  : the description lines can be specified
  Long Description 3  : in combination with the file update
  Long Description 4  : keyword 'NEW'.
  Long Description 5  :
```

Glossary

This glossary includes terms and definitions from:

- The *IBM Dictionary of Computing* New York: McGraw-Hill, 1994.
- The *American National Standard Dictionary for Information Systems* , ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies can be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018. Definitions are identified by the symbol (A) after the definition.
- The *Information Technology Vocabulary* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.

The following cross-references are used in this glossary:

Contrast with. This refers to a term that has an opposed or substantively different meaning.
Deprecated term for. This indicates that the term should not be used. It refers to a preferred term, which is defined in its proper place in the glossary.
See. This refers the reader to multiple-word terms in which this term appears.
See also. This refers the reader to terms that have a related, but not synonymous, meaning.
Synonym for. This indicates that the term has the same meaning as a preferred term, which is defined in the glossary.
Synonymous with. This is a backward reference from a defined term to all other terms that have the same meaning.

A

ACF. Automation control file.

ACF/NCP. Advanced Communications Function for the Network Control Program. See *Advanced Communications Function* and *Network Control Program*.

ACF/VTAM. Advanced Communications Function for the Virtual Telecommunications Access Method. Synonym for *VTAM*. See *Advanced Communications Function* and *Virtual Telecommunications Access Method*.

active monitoring. In SA z/OS, the acquiring of resource status information by soliciting such information at regular, user-defined intervals. See also *passive monitoring*.

adapter. Hardware card that enables a device, such as a workstation, to communicate with another device, such as a monitor, a printer, or some other I/O device.

Address Space Workflow. In RMF, a measure of how a job uses system resources and the speed at which the job moves through the system. A low workflow indicates that a job has few of the resources it needs and is contending with other jobs for system resources. A high workflow indicates that a job has all the resources it needs to execute.

adjacent hosts. Systems connected in a peer relationship using adjacent NetView sessions for purposes of monitoring and control.

adjacent NetView. In SA z/OS, the system defined as the communication path between two SA z/OS systems that do not have a direct link. An adjacent NetView is used for message forwarding and as a communication link between two SA z/OS systems. For example, the adjacent NetView is used when sending responses from a focal point to a remote system.

Advanced Communications Function (ACF). A group of IBM licensed programs (principally VTAM, TCAM, NCP, and SSP) that use the concepts of Systems Network Architecture (SNA), including distribution of function and resource sharing.

advanced program-to-program communication (APPC). A set of inter-program communication services that support cooperative transaction processing in a Systems Network Architecture (SNA) network. APPC is the implementation, on a given system, of SNA's logical unit type 6.2.

alert. (1) In SNA, a record sent to a system problem management focal point or to a collection point to communicate the existence of an alert condition. (2) In NetView, a high-priority event that warrants immediate attention. A database record is generated for certain event types that are defined by user-constructed filters.

alert condition. A problem or impending problem for which some or all of the process of problem determination, diagnosis, and resolution is expected to require action at a control point.

alert focal-point system. See entry for NPDA focal-point system under *focal-point system*.

alert threshold. An application or volume service value that determines the level at which SA z/OS changes the associated icon in the graphical interface to the alert color. SA z/OS may also issue an alert. See *warning threshold*.

AMC. (1) Automation Manager Configuration (2) The Auto Msg Classes entry type

APF. Authorized program facility.

API. Application programming interface.

APPC. Advanced program-to-program communications.

application. An z/OS subsystem or job monitored by SA z/OS.

Application entry. A construct, created with the customization dialogs, used to represent and contain policy for an application.

application group. A named set of applications. An application group is part of an SA z/OS enterprise definition and is used for monitoring purposes.

ApplicationGroup entry. A construct, created with the customization dialogs, used to represent and contain policy for an application group.

application program. (1) A program written for or by a user that applies to the user's work, such as a program that does inventory or payroll. (2) A program used to connect and communicate with stations in a network, enabling users to perform application-oriented activities.

ARM. Automatic restart management.

ASCB. Address space control block.

ASCB status. An application status derived by SA z/OS running a routine (the ASCB checker) that searches the z/OS address space control blocks (ASCBs) for address spaces with a particular job name. The job name used by the ASCB checker is the job name defined in the customization dialog for the application.

ASCII (American National Standard Code for Information Interchange). The standard code, using a coded character set consisting of 7-bit coded characters (8-bit including parity check), for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters. (A)

ASF. Automation status file.

authorized program facility (APF). A facility that permits identification of programs that are authorized to use restricted functions.

automated function. SA z/OS automated functions are automation operators, NetView autotasks that are assigned to perform specific automation functions. However, SA z/OS defines its own synonyms, or *automated function names*, for the NetView autotasks, and these function names are referred to in the sample policy databases provided by SA z/OS. For example, the automation operator AUTBASE corresponds to the SA z/OS automated function BASEOPER.

automated console operations (ACO). The concept (versus a product) of using computers to perform a large subset of tasks ordinarily performed by operators, or assisting operators in performing these tasks.

automatic restart management (ARM). A z/OS recovery function that improves the availability of specified subsystems and applications by automatically restarting them under certain circumstances. Automatic restart management is a function of the Cross-System Coupling Facility (XCF) component of z/OS.

automatic restart management element name. In MVS 5.2 or later, z/OS automatic restart management requires the specification of a unique sixteen character name for each address space that registers with it. All automatic restart management policy is defined in terms of the element name, including SA z/OS's interface with it.

automation. The automatic initiation of actions in response to detected conditions or events. SA z/OS provides automation for z/OS applications, z/OS components, and remote systems that run z/OS. SA z/OS also provides tools that can be used to develop additional automation.

automation agent. In SA z/OS, the automation function is split up between the automation manager and the automation agents. The observing, reacting and doing parts are located within the NetView address space, and are known as the ***automation agents***. The automation agents are responsible for:

- recovery processing
- message processing
- active monitoring: they propagate status changes to the automation manager

automation configuration file. The data set that consists of:

- the automation control file (ACF)
- the automation manager configuration file (AMC)
- the NetView automation table (AT)
- the MPFLSTSA member

automation control file (ACF). In SA z/OS, a file that contains system-level automation policy information. There is one master automation control file for each NetView system on which SA z/OS is installed. Additional policy information and all resource status information is contained in the policy database (PDB). The SA z/OS customization dialogs must be used to build the automation control files. They must not be edited manually.

automation flags. In SA z/OS, the automation policy settings that determine the operator functions that are automated for a resource and the times during which automation is active. When SA z/OS is running, automation is controlled by automation flag policy settings and override settings (if any) entered by the operator. Automation flags are set using the customization dialogs.

automation manager. In SA z/OS, the automation function is split up between the automation manager and the automation agents. The coordination, decision making and controlling functions are processed by each sysplex's ***automation manager***.

The automation manager contains a model of all of the automated resources within the sysplex. The automation agents feed the automation manager with status information and perform the actions that the automation manager tells them to.

The automation manager provides ***sysplex-wide*** automation.

Automation Manager Configuration. The Automation Manager Configuration file (AMC) contains an image of the automated systems in a sysplex or of a standalone system.

Automation NetView. In SA z/OS the NetView that performs routine operator tasks with command procedures or uses other ways of automating system and network management, issuing automatic responses to messages and management services units.

automation operator. NetView automation operators are NetView autotasks that are assigned to perform specific automation functions. See also *automated function*. NetView automation operators may receive messages and process automation procedures. There are no logged-on users associated with automation operators. Each automation operator is an operating system task and runs concurrently with other NetView tasks. An automation operator could be set up to handle JES2 messages that schedule automation procedures, and an automation statement could route such messages to the automation operator. Similar to *operator station task*. SA z/OS message monitor tasks and target control tasks are automation operators.

automation policy. The policy information governing automation for individual systems. This includes automation for applications, z/OS subsystems, z/OS data sets, and z/OS components.

automation policy settings. The automation policy information contained in the automation control file. This information is entered using the customization dialogs. You can display or modify these settings using the customization dialogs.

automation procedure. A sequence of commands, packaged as a NetView command list or a command processor written in a high-level language. An automation procedure performs automation functions and runs under NetView.

automation status file. In SA z/OS, a file containing status information for each automated subsystem, component or data set. This information is used by SA z/OS automation when taking action or when determining what action to take. In Release 2 and above of AOC/MVS, status information is also maintained in the operational information base.

automation table (AT). See *NetView automation table*.

autotask. A NetView automation task that receives messages and processes automation procedures. There are no logged-on users associated with autotasks. Each autotask is an operating system task and runs concurrently with other NetView tasks. An autotask could be set up to handle JES2 messages that schedule automation procedures, and an automation statement could route such messages to the autotasks. Similar to *operator station task*. SA z/OS message monitor tasks and target control tasks are autotasks. Also called *automation operator*.

available. In VTAM programs, pertaining to a logical unit that is active, connected, enabled, and not at its session limit.

B

basic mode. A central processor mode that does not use logical partitioning. Contrast with *logically partitioned (LPAR) mode*.

BCP Internal Interface. Processor function of CMOS-390, zSeries processor families. It allows the communication between basic control programs such as z/OS and the processor support element in order to exchange information or to perform processor control functions. Programs using this function can perform hardware operations such as ACTIVATE or SYSTEM RESET.

beaconing. The repeated transmission of a frame or messages (beacon) by a console or workstation upon detection of a line break or outage.

BookManager. An IBM product that lets users view softcopy documents on their workstations.

C

central processor (CP). The part of the computer that contains the sequencing and processing facilities for instruction execution, initial program load (IPL), and other machine operations.

central processor complex (CPC). A physical collection of hardware that consists of central storage, one or more central processors, timers, and channels.

central site. In a distributed data processing network, the central site is usually defined as the focal point for alerts, application design, and remote system management tasks such as problem management.

CFR/CFS and ISC/ISR. I/O operations can display and return data about integrated system channels (ISC) connected to a coupling facility and coupling facility receiver (CFR) channels and coupling facility sender (CFS) channels.

channel. A path along which signals can be sent; for example, data channel, output channel. See also *link*.

channel path identifier. A system-unique value assigned to each channel path.

CHPID. In SA z/OS, channel path ID; the address of a channel.

CHPID port. A label that describes the system name, logical partitions, and channel paths.

channel-attached. (1) Attached directly by I/O channels to a host processor (for example, a channel-attached device). (2) Attached to a controlling unit by cables, rather than by telecommunication lines. Contrast with *link-attached*. Synonymous with *local*.

CI. Console integration.

CICS/VS. Customer Information Control System for Virtual Storage.

CLIST. Command list.

clone. A set of definitions for application instances that are derived from a basic application definition by substituting a number of different system-specific values into the basic definition.

clone ID. A generic means of handling system-specific values such as the MVS SYSCLONE or the VTAM subarea number. Clone IDs can be substituted into application definitions and commands to customize a basic application definition for the system that it is to be instantiated on.

CNC. A channel path that transfers data between a host system image and an ESCON control unit. It can be point-to-point or switchable.

command. A request for the performance of an operation or the execution of a particular program.

command facility. The component of NetView that is a base for command processors that can monitor, control, automate, and improve the operation of a network. The successor to NCCF.

command list (CLIST). (1) A list of commands and statements, written in the NetView command list language or the REXX language, designed to perform a specific function for the user. In its simplest form, a command list is a list of commands. More complex command lists incorporate variable substitution and conditional logic, making the command list more like a conventional program. Command lists are typically interpreted rather than being compiled. (2) In SA z/OS, REXX command lists that can be used for automation procedures.

command procedure. In NetView, either a command list or a command processor.

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

Command Tree/2. An OS/2-based program that helps you build commands on an OS/2 window, then routes the commands to the destination you specify (such as a 3270 session, a file, a command line, or an application program). It provides the capability for operators to build commands and route them to a specified destination.

common commands. The SA z/OS subset of the CPC operations management commands.

common routine. One of several SA z/OS programs that perform frequently used automation functions. Common routines can be used to create new automation procedures.

Common User Access (CUA) architecture. Guidelines for the dialog between a human and a workstation or terminal.

communication controller. A type of communication control unit whose operations are controlled by one or more programs stored and executed in the unit or by a program executed in a processor to which the controller is connected. It manages the details of line control and the routing of data through a network.

communication line. Deprecated term for *telecommunication line*.

connectivity view. In SA z/OS, a display that uses graphic images for I/O devices and lines to show how they are connected.

console automation. The process of having NetView facilities provide the console input usually handled by the operator.

console connection. In SA z/OS, the 3270 or ASCII (serial) connection between a PS/2 computer and a target system. Through this connection, the workstation appears (to the target system) to be a console.

console integration (CI). A hardware facility that if supported by an operating system, allows operating system messages to be transferred through an internal hardware interface for display on a system console. Conversely, it allows operating system commands entered at a system console to be transferred through an internal hardware interface to the operating system for processing.

consoles. Workstations and 3270-type devices that manage your enterprise.

Control units. Hardware units that control I/O operations for one or more devices. You can view information about control units through I/O operations, and can start or stop data going to them by blocking and unblocking ports.

controller. A unit that controls I/O operations for one or more devices.

couple data set. A data set that is created through the XCF couple data set format utility and, depending on its designated type, is shared by some or all of the z/OS systems in a sysplex. See also *sysplex couple data set* and *XCF couple data set*.

coupling facility. The hardware element that provides high-speed caching, list processing, and locking functions in a sysplex.

CP. Central processor.

CPC. Central processor complex.

CPC operations management commands. A set of commands and responses for controlling the operation of System/390 CPCs.

CPC subset. All or part of a CPC. It contains the minimum *resource* to support a single control program.

CPCB. Command processor control block; an I/O operations internal control block that contains information about the command being processed.

CPU. Central processing unit. Deprecated term for *processor*.

cross-system coupling facility (XCF). XCF is a component of z/OS that provides functions to support cooperation between authorized programs running within a sysplex.

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

Customer Information Control System (CICS). A general-purpose transactional program that controls online communication between terminal users and a database for a large number of end users on a real-time basis.

customization dialogs. The customization dialogs are an ISPF application. They are used to customize the enterprise policy, like, for example, the enterprise resources and the relationships between resources, or the automation policy for systems in the enterprise. How to use these dialogs is described in *IBM Tivoli System Automation for z/OS Customizing and Programming*.

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

D

DASD. Direct access storage device.

data services task (DST). The NetView subtask that gathers, records, and manages data in a VSAM file or a network device that contains network management information.

data set. The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access.

data set members. Members of partitioned data sets that are individually named elements of a larger file that can be retrieved by name.

DBCS. Double-byte character set.

DCCF. Disabled console communication facility.

DCF. Document composition facility.

DELAY Report. An RMF report that shows the activity of each job in the system and the hardware and software resources that are delaying each job.

Devices. You can see information about all devices (such as printers, tape or disk drives, displays, or communications controllers) attached to a particular switch, and control paths and jobs to devices.

DEVR Report. An RMF report that presents information about the activity of I/O devices that are delaying jobs.

dialog. Interactive 3270 panels.

direct access storage device (DASD). A device in which the access time is effectively independent of the location of the data; for example, a disk.

disabled console communication facility (DCCF). A z/OS component that provides limited-function console communication during system recovery situations.

display. (1) To present information for viewing, usually on the screen of a workstation or on a hardcopy device. (2) Deprecated term for *panel*.

disk operating system (DOS). (1) An operating system for computer systems that use disks and diskettes for auxiliary storage of programs and data. (2) Software for a personal computer that controls the processing of programs. For the IBM Personal Computer, the full name is Personal Computer Disk Operating System (PCDOS).

distribution manager. The component of the NetView program that enables the host system to use, send, and delete files and programs in a network of computers.

domain. (1) An access method and its application programs, communication controllers, connecting lines, modems, and attached workstations. (2) In SNA, a system services control point (SSCP) and the physical units (PUs), logical units (LUs), links, link stations, and associated resources that the SSCP can control by means of activation requests and deactivation requests.

double-byte character set (DBCS). A character set, such as Kanji, in which each character is represented by a 2-byte code.

DP enterprise. Data processing enterprise.

DSIPARM. This file is a collection of members of NetView's customization.

DST. Data Services Task.

E

EBCDIC. Extended binary-coded decimal interchange code. A coded character set consisting of 8-bit coded characters.

ECB. Event control block. A control block used to represent the status of an event.

EMCS. Extended multiple console support.

enterprise. An organization, such as a business or a school, that uses data processing.

enterprise monitoring. Enterprise monitoring is used by SA z/OS to update the *NetView Management Console (NMC)* resource status information that is stored in the *Resource Object Data Manager (RODM)*. Resource status information is acquired by enterprise monitoring of the *Resource Measurement Facility (RMF) Monitor III* service information at user-defined intervals. SA z/OS stores this information in its operational information base, where it is used to update the information presented to the operator in graphic displays.

entries. Resources, such as processors, entered on panels.

entry type. Resources, such as processors or applications, used for automation and monitoring.

environment. Data processing enterprise.

error threshold. An automation policy setting that specifies when SA z/OS should stop trying to restart or recover an application, subsystem or component, or offload a data set.

ESA. Enterprise Systems Architecture.

eServer. Processor family group designator used by the SA z/OS customization dialogs to define a target hardware as member of the zSeries or 390-CMOS processor families.

event. (1) In NetView, a record indicating irregularities of operation in physical elements of a network. (2) An occurrence of significance to a task; for example, the completion of an asynchronous operation, such as an input/output operation. (3) Events are part of a trigger condition, in a way that if all events of a trigger condition have occurred, a *STARTUP* or *SHUTDOWN* of an application is performed.

exception condition. An occurrence on a system that is a deviation from normal operation. SA z/OS monitoring highlights exception conditions and allows an SA z/OS enterprise to be managed by exception.

extended recovery facility (XRF). A facility that minimizes the effect of failures in z/OS, VTAM, the host processor, or high availability applications during sessions between high availability applications and designated terminals. This facility provides an alternate subsystem to take over sessions from the failing subsystem.

F

fallback system. See *secondary system*.

field. A collection of bytes within a record that are logically related and are processed as a unit.

file manager commands. A set of SA z/OS commands that read data from or write data to the automation control file or the operational information base. These commands are useful in the development of automation that uses SA z/OS facilities.

focal point. In NetView, the focal-point domain is the central host domain. It is the central control point for any management services element containing control of the network management data.

focus host. A processor with the role in the context of a unified system image

focal point system. (1) A system that can administer, manage, or control one or more target systems. There are a number of different focal point system associated with IBM automation products. (2) **NMC focal point system**. The NMC focal point system is a NetView system with an attached workstation server and LAN that gathers information about the state of the network. This focal point system uses RODM to store the data it collects in the data model. The information stored in RODM can be accessed from any LAN-connected workstation with NetView Management Console installed. (3) **NPDA focal point system.** This is a NetView system that collects all the NPDA alerts that are generated within your enterprise. It is supported by NetView. If you have SA z/OS installed the NPDA focal point system must be the same as your NMC focal point system. The NPDA focal point system is also known as the *alert focal point system*. (4) **SA z/OS Processor Operations focal point system.** This is a NetView system that has SA z/OS host code installed. The SA z/OS Processor Operations focal point system receives messages from the systems and operator consoles of the machines that it controls. It provides full systems and operations console function for its target systems. It can be used to IPL these systems. Note that some restrictions apply to the Hardware Management Console for an S/390 microprocessor cluster. (5) **SA z/OS SDF focal point system.** The SA z/OS SDF focal point system is an SA z/OS NetView system that collects status information from other SA z/OS NetViews within your enterprise. (6) **Status focal point system.** In NetView, the system to which STATMON, VTAM and NLDM send status information on network resources. If you have a NMC focal point, it must be on the same system as the Status focal point. (7) **Hardware Management Console.** Although not listed as a focal point, the Hardware Management Console acts as a focal point for the console functions of an S/390 microprocessor cluster. Unlike all the other focal points in this definition, the Hardware Management Console runs on a LAN-connected workstation,

frame. For a System/390 microprocessor cluster, a frame contains one or two central processor complexes (CPCs), support elements, and AC power distribution.

full-screen mode. In NetView, a form of panel presentation that makes it possible to display the contents of an entire workstation screen at once. Full-screen mode can be used for fill-in-the-blanks prompting. Contrast with *line mode*.

G

gateway session. An NetView-NetView Task session with another system in which the SA z/OS outbound gateway operator logs onto the other NetView session without human operator intervention. Each end of a gateway session has both an inbound and outbound gateway operator.

generic alert. Encoded alert information that uses code points (defined by IBM and possibly customized by users or application programs) stored at an alert receiver, such as NetView.

generic routines. In SA z/OS, a set of self-contained automation routines that can be called from the NetView automation table, or from user-written automation procedures.

group. A collection of target systems defined through configuration dialogs. An installation might set up a group to refer to a physical site or an organizational or application entity.

group entry. A construct, created with the customization dialogs, used to represent and contain policy for a group.

group entry type. A collection of target systems defined through the customization dialog. An installation might set up a group to refer to a physical site or an organizational entity. Groups can, for example, be of type STANDARD or SYSPLEX.

H

Hardware Management Console. A console used by the operator to monitor and control a System/390 microprocessor cluster.

Hardware Management Console Application (HWMCA). A direct-manipulation object-oriented graphical user interface that provides single point of control and single system image for hardware elements. HWMCA provides customer grouping support, aggregated and real-time system status using colors, consolidated hardware messages support, consolidated operating system messages support, consolidated service support, and hardware commands targeted at a single system, multiple systems, or a customer group of systems.

heartbeat. In SA z/OS, a function that monitors the validity of the status forwarding path between remote systems and the NMC focal point, and monitors the availability of remote z/OS systems, to ensure that status information displayed on the SA z/OS workstation is current.

help panel. An online panel that tells you how to use a command or another aspect of a product.

hierarchy. In the NetView program, the resource types, display types, and data types that make up the organization, or levels, in a network.

high-level language (HLL). A programming language that does not reflect the structure of any particular computer or operating system. For the NetView program, the high-level languages are PL/I and C.

HLL. High-level language.

host system. In a coupled system or distributed system environment, the system on which the facilities for centralized automation run. SA z/OS publications refer to target systems or focal-point systems instead of hosts.

host (primary processor). The processor at which you enter a command (also known as the *issuing processor*).

HWMCA. Hardware Management Console Application. Application for the graphic hardware management console that monitors and controls a central processor complex. It is attached to a target processor (a system 390 microprocessor cluster) as a dedicated system console. This microprocessor uses OCF to process commands.

I

images. A grouping of processors and I/O devices that you define. You can define a single-image mode that allows a multiprocessor system to function as one central processor image.

IMS/VS. Information Management System/Virtual Storage.

inbound. In SA z/OS, messages sent to the focal-point system from the PC or target system.

inbound gateway operator. The automation operator that receives incoming messages, commands, and responses from the outbound gateway operator at the sending system. The inbound gateway operator handles communications with other systems using a gateway session.

Information Management System/Virtual Storage (IMS/VS). A database/data communication (DB/DC) system that can manage complex databases and networks. Synonymous with IMS.

INGEIO PROC. The I/O operations default procedure name; part of the SYS1.PROCLIB.

initial program load (IPL). (1) The initialization procedure that causes an operating system to commence operation. (2) The process by which a configuration image is loaded into storage at the beginning of a workday or after a system malfunction. (3) The process of loading system programs and preparing a system to run jobs.

initialize automation. SA z/OS-provided automation that issues the correct z/OS start command for each subsystem when SA z/OS is initialized. The automation ensures that subsystems are started in the order specified in the automation control file and that prerequisite applications are functional.

input/output support processor (IOSP). The hardware unit that provides I/O support functions for the primary support processor and maintenance support functions for the processor controller.

Interactive System Productivity Facility (ISPF). An IBM licensed program that serves as a full-screen editor and dialog manager. Used for writing application programs, it provides a means of generating standard screen panels and interactive dialogs between the application programmer and the terminal user.

interested operator list. The list of operators who are to receive messages from a specific target system.

internal token. A *logical token* (LTOK); name by which the I/O resource or object is known; stored in IODF.

IOCDS. I/O configuration data set. The data set that describes the I/O configuration.

I/O Ops. I/O operations.

IOSP. Input/Output Support Processor.

I/O operations. The part of SA z/OS that provides you with a single point of logical control for managing connectivity in your active I/O configurations. I/O operations takes an active role in detecting unusual conditions and lets you view and change paths between a processor and an I/O device, using dynamic switching (the ESCON director). Also known as I/O Ops.

I/O resource number. Combination of channel path identifier (CHPID), device number, etc. See internal token.

IPL. Initial program load.

ISA. Industry Standard Architecture.

ISPF. Interactive System Productivity Facility.

ISPF console. From this 3270-type console you are logged onto ISPF to use the runtime panels for I/O operations and SA z/OS customization panels.

issuing host. See *primary host*; the base program at which you enter a command for processing.

J

JCL. Job control language.

JES. Job entry subsystem.

job. (1) A set of data that completely defines a unit of work for a computer. A job usually includes all necessary computer programs, linkages, files, and instructions to the operating system. (2) An address space.

job control language (JCL). A problem-oriented language designed to express statements in a job that are used to identify the job or describe its requirements to an operating system.

job entry subsystem (JES). A facility for spooling, job queuing, and managing I/O. In SA z/OS publications, JES refers to JES2 or JES3, unless distinguished as being either one or the other.

K

Kanji. An ideographic character set used in Japanese. See also *double-byte character set*.

L

LAN. Local area network.

line mode. A form of screen presentation in which the information is presented a line at a time in the message area of the terminal screen. Contrast with *full-screen mode*.

link. (1) In SNA, the combination of the link connection and the link stations joining network nodes; for example, a System/370 channel and its associated protocols, a serial-by-bit connection under the control of synchronous data link control (SDLC). (2) In SA z/OS, link connection is the physical medium of transmission.

link-attached. Describes devices that are physically connected by a telecommunication line. Contrast with *channel-attached*.

Linux for zSeries and S/390. UNIX-like open source operating system conceived by Linus Torvalds and developed across the internet.

local. Pertaining to a device accessed directly without use of a telecommunication line. Synonymous with *channel-attached*.

local area network (LAN). (1) A network in which a set of devices is connected for communication. They can be connected to a larger network. See also *token ring*. (2) A network in which communications are limited to a moderately-sized geographic area such as a single office building, warehouse, or campus, and that do not generally extend across public rights-of-way.

logical partition (LP). A subset of the processor hardware that is defined to support an operating system. See also *logically partitioned (LPAR) mode*.

logical switch number (LSN). Assigned with the switch parameter of the CHPID macro of the IOCP.

logical token (LTOK). Resource number of an object in the IODF.

logical unit (LU). In SNA, a port through which an end user accesses the SNA network and the functions provided by system services control points (SSCPs). An LU can support at least two sessions — one with an SSCP and one with another LU — and may be capable of supporting many sessions with other LUs. See also *physical unit (PU)* and *system services control point (SSCP)*.

logical unit (LU) 6.2. A type of logical unit that supports general communications between programs in a distributed processing environment. LU 6.2 is characterized by (a) a peer relationship between session partners, (b) efficient use of a session for multiple transactions, (c) comprehensive end-to-end error processing, and (d) a generic application program interface (API) consisting of structured verbs that are mapped into a product implementation. Synonym for advanced program-to-program communications (APPC).

logically partitioned (LPAR) mode. A central processor mode that enables an operator to allocate system processor hardware resources among several logical partitions. Contrast with *basic mode*.

LOGR. The sysplex logger.

LP. Logical partition.

LPAR. Logically partitioned (mode).

LU. Logical unit.

LU-LU session. In SNA, a session between two logical units (LUs) in an SNA network. It provides communication between two end users, or between an end user and an LU services component.

LU 6.2. Logical unit 6.2.

LU 6.2 session. A session initiated by VTAM on behalf of an LU 6.2 application program, or a session initiated by a remote LU in which the application program specifies that VTAM is to control the session by using the APPCCMD macro.

M

MAT. Deprecated term for NetView Automation Table.

MCA. Micro Channel* architecture.

MCS. Multiple console support.

member. A specific function (one or more modules/routines) of a multisystem application that is defined to XCF and assigned to a group by the multisystem application. A member resides on one system in the sysplex and can use XCF services to communicate (send and receive data) with other members of the same group.

message automation table (MAT). Deprecated term for NetView Automation Table.

message class. A number that SA z/OS associates with a message to control routing of the message. During automated operations, the classes associated with each message issued by SA z/OS are compared to the classes assigned to each notification operator. Any operator with a class matching one of the message's classes receives the message.

message forwarding. The SA z/OS process of sending messages generated at an SA z/OS target system to the SA z/OS focal-point system.

message group. Several messages that are displayed together as a unit.

message monitor task. A task that starts and is associated with a number of communications tasks. Message monitor tasks receive inbound messages from a communications task, determine the originating target system, and route the messages to the appropriate target control tasks.

message processing facility (MPF). A z/OS table that screens all messages sent to the z/OS console. The MPF compares these messages with a customer-defined list of messages on which to automate, suppress from the z/OS console display, or both, and marks messages to automate or suppress. Messages are then broadcast on the subsystem interface (SSI).

message suppression. The ability to restrict the amount of message traffic displayed on the z/OS console.

Micro Channel architecture. The rules that define how subsystems and adapters use the Micro Channel bus in a computer. The architecture defines the services that each subsystem can or must provide.

microprocessor. A processor implemented on one or a small number of chips.

migration. Installation of a new version or release of a program to replace an earlier version or release.

MP. Multiprocessor.

MPF. Message processing facility.

MPFLSTSA. The MPFLST member that is built by SA z/OS.

Multiple Virtual Storage (MVS). An IBM licensed program. MVS, which is the predecessor of OS/390, is an operating system that controls the running of programs on a System/390 or System/370 processor. MVS includes an appropriate level of the Data Facility Product (DFP) and Multiple Virtual Storage/Enterprise Systems Architecture System Product Version 5 (MVS/ESA SP5).

multiprocessor (MP). A CPC that can be physically partitioned to form two operating processor complexes.

multisystem application. An application program that has various functions distributed across z/OS images in a multisystem environment.

multisystem environment. An environment in which two or more z/OS images reside in one or more processors, and programs on one image can communication with programs on the other images.

MVS. Multiple Virtual Storage, predecessor of z/OS.

MVS image. A single occurrence of the MVS/ESA operating system that has the ability to process work.

MVS/JES2. Multiple Virtual Storage/Job Entry System 2. A z/OS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In an installation with more than one processor, each JES2 processor independently controls its job input, scheduling, and output processing.

MVS/ESA. Multiple Virtual Storage/Enterprise Systems Architecture.

N

NAU. (1) Network accessible unit. (2) Network addressable unit.

NCCF. Network Communications Control Facility.

NCP. (1) Network Control Program (IBM licensed program). Its full name is Advanced Communications Function for the Network Control Program. Synonymous with *ACF/NCP*. (2) Network control program (general term).

NetView. An IBM licensed program used to monitor a network, manage it, and diagnose network problems. NetView consists of a command facility that includes a presentation service, command processors, automation based on command lists, and a transaction processing structure on which the session monitor, hardware monitor, and terminal access facility (TAF) network management applications are built.

network accessible unit (NAU). A logical unit (LU), physical unit (PU), control point (CP), or system services control point (SSCP). It is the origin or the destination of information transmitted by the path control network. Synonymous with *network addressable unit*.

network addressable unit (NAU). Synonym for *network accessible unit*.

NetView automation procedures. A sequence of commands, packaged as a NetView command list or a command processor written in a high-level language. An automation procedure performs automation functions and runs under the NetView program.

NetView automation table (AT). A table against which the NetView program compares incoming messages. A match with an entry triggers the specified response. SA z/OS entries in the NetView automation table trigger an SA z/OS response to target system conditions. Formerly known as the message automation table (MAT).

NetView Command list language. An interpretive language unique to NetView that is used to write command lists.

NetView (NCCF) console. A 3270-type console for NetView commands and runtime panels for system operations and processor operations.

NetView Graphic Monitor Facility (NGMF). Deprecated term for NetView Management Console.

NetView hardware monitor. The component of NetView that helps identify network problems, such as hardware, software, and microcode, from a central control point using interactive display techniques. Formerly called *network problem determination application*.

NetView log. The log in which NetView records events pertaining to NetView and SA z/OS activities.

NetView message table. See *NetView automation table*.

NetView Management Console (NMC). A function of the NetView program that provides a graphic, topological presentation of a network that is controlled by the NetView program. It provides the operator different views of a network, multiple levels of graphical detail, and dynamic resource status of the network. This function consists of a series of graphic windows that allows you to manage the network interactively. Formerly known as the NetView Graphic Monitor Facility (NGMF).

NetView-NetView task (NNT). The task under which a cross-domain NetView operator session runs. Each NetView program must have a NetView-NetView task to establish one NNT session. See also *operator station task*.

NetView-NetView Task session. A session between two NetView programs that runs under a NetView-NetView Task. In SA z/OS, NetView-NetView Task sessions are used for communication between focal point and remote systems.

NetView paths via logical unit (LU 6.2). A type of network-accessible port (VTAM connection) that enables end users to gain access to SNA network resources and communicate with each other. LU 6.2 permits communication between processor operations and the workstation.

network. (1) An interconnected group of nodes. (2) In data processing, a user application network. See *SNA network*.

Network Communications Control Facility (NCCF). The operations control facility for the network. NCCF consists of a presentation service, command processors, automation based on command lists, and a transaction processing structure on which the network management applications NLDM and NPDA are built. NCCF is a precursor to the NetView command facility.

Network Control Program (NCP). An IBM licensed program that provides communication controller support for single-domain, multiple-domain, and interconnected network capability. Its full name is Advanced Communications Function for the Network Control Program.

Networking NetView. In SA z/OS the NetView that performs network management functions, such as managing the configuration of a network. In SA z/OS it is common to also route alerts to the Networking NetView.

Network Problem Determination Application (NPDA). An NCCF application that helps you identify network problems, such as hardware, software, and microcode, from a central control point using interactive display methods. The alert manager for the network. The precursor of the NetView hardware monitor.

NGMF. Deprecated term for NetView Management Console.

NGMF focal-point system. Deprecated term for NMC focal point system.

NIP. Nucleus initialization program.

NMC focal point system. See *focal point system*

NMC workstation. The NMC workstation is the primary way to dynamically monitor SA z/OS systems. From the windows, you see messages, monitor status, view trends, and react to changes before they cause problems for end users. You can use multiple windows to monitor multiple views of the system.

NNT. NetView-NetView task.

notification message. An SA z/OS message sent to a human notification operator to provide information about significant automation actions. Notification messages are defined using the customization dialogs.

notification operator. A NetView console operator who is authorized to receive SA z/OS notification messages. Authorization is made through the customization dialogs.

NPDA. Network Problem Determination Application.

NPDA focal-point system. See *focal-point system*.

NTRI. NCP/token-ring interconnection.

nucleus initialization program (NIP). The program that initializes the resident control program; it allows the operator to request last-minute changes to certain options specified during system generation.

O

objective value. An average Workflow or Using value that SA z/OS can calculate for applications from past service data. SA z/OS uses the objective value to calculate warning and alert thresholds when none are explicitly defined.

OCA. In SA z/OS, operator console A, the active operator console for a target system. Contrast with *OCB*.

OCB. In SA z/OS, operator console B, the backup operator console for a target system. Contrast with *OCA*.

OCF. Operations command facility.

OCF-based processor. A central processor complex that uses an operations command facility for interacting with human operators or external programs to perform operations management functions on the CPC.

OPC/A. Operations Planning and Control/Advanced.

OPC/ESA. Operations Planning and Control/Enterprise Systems Architecture.

operating system (OS). Software that controls the execution of programs and that may provide services such as resource allocation, scheduling, input/output control, and data management. Although operating systems are predominantly software, partial hardware implementations are possible. (T)

operations. The real-time control of a hardware device or software function.

operations command facility (OCF). A facility of the central processor complex that accepts and processes operations management commands.

Operations Planning and Control/Advanced (OPC/A). A set of IBM licensed programs that automate, plan, and control batch workload. OPC/A analyzes system and workload status and submits jobs accordingly.

Operations Planning and Control/ESA (OPC/ESA). A set of IBM licensed programs that automate, plan, and control batch workload. OPC/ESA analyzes system and workload status and submits jobs accordingly. The successor to OPC/A.

operator. (1) A person who keeps a system running. (2) A person or program responsible for managing activities controlled by a given piece of software such as z/OS, the NetView program, or IMS. (3) A person who operates a device. (4) In a language statement, the lexical entity that indicates the action to be performed on operands.

operator console. (1) A functional unit containing devices that are used for communications between a computer operator and a computer. (T) (2) A display console used for communication between the operator and the system, used primarily to specify information concerning application programs and I/O operations and to monitor system operation. (3) In SA z/OS, a console that displays output from and sends input to the operating system (z/OS, LINUX, VM, VSE). Also called *operating system console*. In the SA z/OS operator commands and configuration dialogs, OC is used to designate a target system operator console.

operator station task (OST). The NetView task that establishes and maintains the online session with the network operator. There is one operator station task for each network operator who logs on to the NetView program.

operator view. A set of group, system, and resource definitions that are associated together for monitoring purposes. An operator view appears as a graphic display in the graphical interface showing the status of the defined groups, systems, and resources.

OperatorView entry. A construct, created with the customization dialogs, used to represent and contain policy for an operator view.

OS. Operating system.

z/OS component. A part of z/OS that performs a specific z/OS function. In SA z/OS, component refers to entities that are managed by SA z/OS automation.

z/OS subsystem. Software products that augment the z/OS operating system. JES and TSO/E are examples of z/OS subsystems. SA z/OS includes automation for some z/OS subsystems.

z/OS system. A z/OS image together with its associated hardware, which collectively are often referred to simply as a system, or z/OS system.

OSA. I/O operations can display the open system adapter (OSA) channel logical definition, physical attachment, and status. You can configure an OSA channel on or off.

OST. Operator station task.

outbound. In SA z/OS, messages or commands from the focal-point system to the target system.

outbound gateway operator. The automation operator that establishes connections to other systems. The outbound gateway operator handles communications with other systems through a gateway session. The automation operator sends messages, commands, and responses to the inbound gateway operator at the receiving system.

P

page. (1) The portion of a panel that is shown on a display surface at one time. (2) To transfer instructions, data, or both between real storage and external page or auxiliary storage.

panel. (1) A formatted display of information that appears on a terminal screen. Panels are full-screen 3270-type displays with a monospaced font, limited color and graphics. (2) By using SA z/OS panels you can see status, type commands on a command line using a keyboard, configure your system, and passthru to other consoles. See also *help panel*. (3) In computer graphics, a display image that defines the locations and characteristics of display fields on a display surface. Contrast with *screen*.

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

parameter. (1) A variable that is given a constant value for a specified application and that may denote the application. (2) An item in a menu for which the user specifies a value or for which the system provides a value when the menu is interpreted. (3) Data passed to a program or procedure by a user or another program, namely as an operand in a language statement, as an item in a menu, or as a shared data structure.

partition. (1) A fixed-size division of storage. (2) In VSE, a division of the virtual address area that is available for program processing. (3) On an IBM Personal Computer fixed disk, one of four possible storage areas of variable size; one can be accessed by DOS, and each of the others may be assigned to another operating system.

partitionable CPC. A CPC that can be divided into 2 independent CPCs. See also *physical partition, single-image mode, MP, side*.

partitioned data set (PDS). A data set in direct access storage that is divided into partitions, called *members*, each of which can contain a program, part of a program, or data.

passive monitoring. In SA z/OS, the receiving of unsolicited messages from z/OS systems and their resources. These messages can prompt updates to resource status displays. See also *active monitoring.*

PCE. Processor controller. Also known as the “support processor” or “service processor” in some processor families.

PDB. Policy Database

PDS. Partitioned data set.

physical partition. Part of a CPC that operates as a CPC in its own right, with its own copy of the operating system.

physical unit (PU). In SNA, the component that manages and monitors the resources (such as attached links and adjacent link stations) of a node, as requested by a system services control point (SSCP) through an SSCP-PU session. An SSCP activates a session with the physical unit to indirectly manage, through the PU, resources of the node such as attached links.

physically partitioned (PP) configuration. A mode of operation that allows a multiprocessor (MP) system to function as two or more independent CPCs having separate power, water, and maintenance boundaries. Contrast with *single-image (SI) configuration*.

POI. Program operator interface.

policy. The automation and monitoring specifications for an SA z/OS enterprise. See *IBM Tivoli System Automation for z/OS Defining Automation Policy*.

policy database. The database where the automation policy is recorded. Also known as the PDB.

POR. Power-on reset.

port. (1) System hardware to which the I/O devices are attached. (2) On an ESCON switch, a port is an addressable connection. The switch routes data through the ports to the channel or control unit. Each port has a name that can be entered into a switch matrix, and you can use commands to change the switch configuration. (3) An access point (for example, a logical unit) for data entry or exit. (4) A functional unit of a node through which data can enter or leave a data network. (5) In data communication, that part of a data processor that is dedicated to a single data channel for the purpose of receiving data from or transmitting data to one or more external, remote devices. (6) power-on reset (POR) (7) A function that re-initializes all the hardware in a CPC and loads the internal code that enables the CPC to load and run an operating system.

PP. Physically partitioned (configuration).

PPT. Primary POI task.

primary host. The base program at which you enter a command for processing.

primary POI task (PPT). The NetView subtask that processes all unsolicited messages received from the VTAM program operator interface (POI) and delivers them to the controlling operator or to the command processor. The PPT also processes the initial command specified to execute when NetView is initialized and timer request commands scheduled to execute under the PPT.

primary system. A system is a primary system for an application if the application is normally meant to be running there. SA z/OS starts the application on all the primary systems defined for it.

problem determination. The process of determining the source of a problem; for example, a program component, machine failure, telecommunication facilities, user or contractor-installed programs or equipment, environment failure such as a power loss, or user error.

processor controller. Hardware that provides support and diagnostic functions for the central processors.

processor operations. The part of SA z/OS that monitors and controls processor (hardware) operations. Processor operations provides a connection from a focal-point system to a target system. Through NetView on the focal-point system, processor operations automates operator and system consoles for monitoring and recovering target systems. Also known as ProcOps.

processor operations control file. Named by your system programmer, this file contains configuration and customization information. The programmer records the name of this control file in the processor operations file generation panel ISQDPG01.

Processor Resource/Systems Manager (PR/SM). The feature that allows the processor to use several operating system images simultaneously and provides logical partitioning capability. See also *LPAR*.

ProcOps. Processor operations.

ProcOps Service Machine (PSM). The PSM is a CMS user on a VM host system. It runs a CMS multitasking application that serves as ″virtual hardware″ for ProcOps. ProOps communicates via the PSM with the VM guest systems that are defined as target systems within ProcOps.

product automation. Automation integrated into the base of SA z/OS for the products DB2, CICS, IMS, OPC (formerly called *features*).

program to program interface (PPI). A NetView function that allows user programs to send or receive data buffers from other user programs and to send alerts to the NetView hardware monitor from system and application programs.

protocol. In SNA, the meanings of, and the sequencing rules for, requests and responses used for managing the network, transferring data, and synchronizing the states of network components.

proxy resource. A resource defined like an entry type APL representing a processor operations target system.

PR/SM. Processor Resource/Systems Manager.

PSM. ProcOps Service Machine.

PU. Physical unit.

R

remote system. A system that receives resource status information from an SA z/OS focal-point system. An SA z/OS remote system is defined as part of the same SA z/OS enterprise as the SA z/OS focal-point system to which it is related.

requester. A requester is a workstation software, which enables users to log on to a domain, that is, to the server(s) belonging to this domain, and use the resources in this domain. After the log on to a domain, users can access the shared resources and use the processing capability of the server(s). Because the bigger part of shared resources is on the server(s), users can reduce hardware investment.

resource. (1) Any facility of the computing system or operating system required by a job or task, and including main storage, input/output devices, the processing unit, data sets, and control or processing programs. (2) In NetView, any hardware or software that provides function to the network. (3) In SA z/OS, any z/OS application, z/OS component, job, device, or target system capable of being monitored or automated through SA z/OS.

Resource Access Control Facility (RACF). A program that can provide data security for all your resources. RACF protects data from accidental or deliberate unauthorized disclosure, modification, or destruction.

resource group. A physically partitionable portion of a processor. Also known as a *side*.

Resource Monitoring Facility (RMF) Monitor III. A program that measures and reports on the availability and activity of system hardware and software resources, such as processors, devices, storage, and address spaces. RMF can issue online reports about system performance problems as they occur.

Resource Object Data Manager (RODM). A data cache manager designed to support process control and automation applications. RODM provides an in-memory data cache for maintaining real-time data in an address space that is accessible by multiple applications. RODM also allows an application to query an object and receive a rapid response and act on it.

resource token. A unique internal identifier of an ESCON resource or resource number of the object in the IODF.

restart automation. SA z/OS-provided automation that monitors subsystems to ensure that they are running. If a subsystem fails, SA z/OS attempts to restart it according to the policy in the automation control file.

Restructured Extended Executor (REXX). An interpretive language used to write command lists.

return code. A code returned from a program used to influence the issuing of subsequent instructions.

REXX. Restructured Extended Executor.

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

RMF. Resource Measurement Facility.

RODM. Resource Object Data Manager.

S

SAF. Security Authorization Facility.

SA IOM. System Automation for Integrated Operations Management

SA z/OS. System Automation for z/OS

SA z/OS customization dialogs. An ISPF application through which the SA z/OS policy administrator defines policy for individual z/OS systems and builds automation control data and RODM load function files.

SA z/OS customization focal point system. See *focal point system*.

SA z/OS data model. The set of objects, classes and entity relationships necessary to support the function of SA z/OS and the NetView automation platform.

SA z/OS enterprise. The group of systems and resources defined in the customization dialogs under one enterprise name. An SA z/OS enterprise consists of connected z/OS systems running SA z/OS.

SA z/OS focal point system. See *focal point system*.

SA z/OS policy. The description of the systems and resources that make up an SA z/OS enterprise, together with their monitoring and automation definitions.

SA z/OS policy administrator. The member of the operations staff who is responsible for defining SA z/OS policy.

SA z/OS satellite. If you are running two NetViews on an z/OS system to split the automation and networking functions of NetView, it is common to route alerts to the Networking NetView. For SA z/OS to process alerts properly on the Networking NetView, you must install a subset of SA z/OS code, called an *SA z/OS satellite* on the Networking NetView.

SA z/OS SDF focal point system. See *focal point system*.

SCA. In SA z/OS, system console A, the active system console for a target hardware. Contrast with *SCB*.

SCB. In SA z/OS, system console B, the backup system console for a target hardware. Contrast with *SCA*.

screen. Deprecated term for display panel.

screen handler. In SA z/OS, software that interprets all data to and from a full-screen image of a target system. The interpretation depends on the format of the data on the full-screen image. Every processor and operating system has its own format for the full-screen image. A screen handler controls one PS/2 connection to a target system.

SDF. Status Display Facility.

SDLC. Synchronous data link control.

SDSF. System Display and Search Facility.

secondary system. A system is a secondary system for an application if it is defined to automation on that system, but the application is not normally meant to be running there. Secondary systems are systems to which an application can be moved in the event that one or more of its primary systems are unavailable. SA z/OS does not start the application on its secondary systems.

server. A server is a workstation that shares resources, which include directories, printers, serial devices, and computing powers.

service language command (SLC). The line-oriented command language of processor controllers or service processors.

service processor (SVP). The name given to a processor controller on smaller System/370 processors.

service period. Service periods allow the users to schedule the availability of applications. A service period is a set of time intervals (service windows), during which an application should be active.

service threshold. An SA z/OS policy setting that determines when to notify the operator of deteriorating service for a resource. See also *alert threshold* and *warning threshold*.

session. In SNA, a logical connection between two network addressable units (NAUs) that can be activated, tailored to provide various protocols, and deactivated, as requested. Each session is uniquely identified in a transmission header by a pair of network addresses identifying the origin and destination NAUs of any transmissions exchanged during the session.

session monitor. The component of the NetView program that collects and correlates session-related data and provides online access to this information. The successor to NLDM.

shutdown automation. SA z/OS-provided automation that manages the shutdown process for subsystems by issuing shutdown commands and responding to prompts for additional information.

side. A part of a partitionable CPC that can run as a physical partition and is typically referred to as the A-side or the B-side.

Simple Network Management Protocol (SNMP). An IP based industry standard protocol to monitor and control resources in an IP network.

single image. A processor system capable of being physically partitioned that has not been physically partitioned. Single-image systems can be target hardware processors.

single-image (SI) mode. A mode of operation for a multiprocessor (MP) system that allows it to function as one CPC. By definition, a uniprocessor (UP) operates in single-image mode. Contrast with *physically partitioned (PP) configuration*.

SLC. Service language command.

SMP/E. System Modification Program Extended.

SNA. Systems Network Architecture.

SNA network. In SNA, the part of a user-application network that conforms to the formats and protocols of systems network architecture. It enables reliable transfer of data among end users and provides protocols for controlling the resources of various network configurations. The SNA network consists of network addressable units (NAUs), boundary function components, and the path control network.

SNMP. Simple Network Management Protocol (a TCP/IP protocol). A protocol that allows network management by elements, such as gateways, routers, and hosts. This protocol provides a means of communication between network elements regarding network resources.

solicited message. An SA z/OS message that directly responds to a command. Contrast with *unsolicited message*.

SSCP. System services control point.

SSI. Subsystem interface.

start automation. SA z/OS-provided automation that manages and completes the startup process for subsystems. During this process, SA z/OS replies to prompts for additional information, ensures that the startup process completes within specified time limits, notifies the operator of problems, if necessary, and brings subsystems to an UP (or ready) state.

startup. The point in time at which a subsystem or application is started.

status. The measure of the condition or availability of the resource.

status focal-point system. See *focal-point system*.

status display facility (SDF). The system operations part of SA z/OS that displays status of resources such as applications, gateways, and write-to-operator messages (WTORs) on dynamic color-coded panels. SDF shows spool usage problems and resource data from multiple systems.

steady state automation. The routine monitoring, both for presence and performance, of subsystems, applications, volumes and systems. Steady state automation may respond to messages, performance exceptions and discrepancies between its model of the system and reality.

structure. A construct used by z/OS to map and manage storage on a coupling facility. See cache structure, list structure, and lock structure.

subgroup. A named set of systems. A subgroup is part of an SA z/OS enterprise definition and is used for monitoring purposes.

SubGroup entry. A construct, created with the customization dialogs, used to represent and contain policy for a subgroup.

subplex. Situations where the physical sysplex has been divided into subentities, for example, a test sysplex and a production sysplex. This may be done to isolate the test environment from the production environment.

subsystem. (1) A secondary or subordinate system, usually capable of operating independent of, or asynchronously with, a controlling system. (2) In SA z/OS, an z/OS application or subsystem defined to SA z/OS.

subsystem interface. The z/OS interface over which all messages sent to the z/OS console are broadcast.

support element. A hardware unit that provides communications, monitoring, and diagnostic functions to a central processor complex (CPC).

support processor. Another name given to a processor controller on smaller System/370 processors; see *service processor*.

SVP. Service processor.

switches. ESCON directors are electronic units with ports that dynamically switch to route data to I/O devices. The switches are controlled by I/O operations commands that you enter on a workstation.

switch identifier. The switch device number (swchdevn), the logical switch number (LSN) and the switch name

symbolic destination name (SDN). Used locally at the workstation to relate to the VTAM application name.

synchronous data link control (SDLC). A discipline for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges may be duplex or half-duplex over switched or nonswitched links. The configuration of the link connection may be point-to-point, multipoint, or loop. SDLC conforms to subsets of the Advanced Data Communication Control Procedures (ADCCP) of the American National Standards Institute and High-Level Data Link Control (HDLC) of the International Standards Organization.

SYSINFO Report. An RMF report that presents an overview of the system, its workload, and the total number of jobs using resources or delayed for resources.

SysOps. System operations.

sysplex. A set of z/OS systems communicating and cooperating with each other through certain multisystem hardware components (coupling devices and timers) and software services (couple data sets).

In a sysplex, z/OS provides the coupling services that handle the messages, data, and status for the parts of a multisystem application that has its workload spread across two or more of the connected processors, sysplex timers, coupling facilities, and couple data sets (which contains policy and states for automation).

A Parallel Sysplex is a sysplex that includes a coupling facility.

sysplex application group. A sysplex application group is a grouping of applications that can run on any system in a sysplex.

sysplex couple data set. A couple data set that contains sysplex-wide data about systems, groups, and members that use XCF services. All z/OS systems in a sysplex must have connectivity to the sysplex couple data set. See also *couple data set*.

Sysplex Timer. An IBM unit that synchronizes the time-of-day (TOD) clocks in multiple processors or processor sides. External Time Reference (ETR) is the z/OS generic name for the IBM Sysplex Timer (9037).

system. In SA z/OS, system means a focal point system (z/OS) or a target system (MVS, VM, VSE, LINUX, or CF).

System Automation for Integrated Operations Management. (1) An outboard automation solution for secure remote access to mainframe/distributed systems. Tivoli System Automation for Integrated Operations Management, previously Tivoli AF/REMOTE, allows users to manage mainframe and distributed systems from any location. (2) The full name for SA IOM.

System Automation for OS/390. The full name for SA OS/390, the predecessor to System Automation for z/OS.

System Automation for z/OS. The full name for SA z/OS.

system console. (1) A console, usually having a keyboard and a display screen, that is used by an operator to control and communicate with a system. (2) A logical device used for the operation and control of hardware functions (for example, IPL, alter/display, and reconfiguration). The system console can be assigned to any of the physical displays attached to a processor controller or support processor. (3) In SA z/OS, the hardware system console for processor controllers or service processors of processors connected using SA z/OS. In the SA z/OS operator commands and configuration dialogs, SC is used to designate the system console for a target hardware processor.

System Display and Search Facility (SDSF). An IBM licensed program that provides information about jobs, queues, and printers running under JES2 on a series of panels. Under SA z/OS you can select SDSF from a pull-down menu to see the resources' status, view the z/OS system log, see WTOR messages, and see active jobs on the system.

System entry. A construct, created with the customization dialogs, used to represent and contain policy for a system.

System Modification Program/Extended (SMP/E). An IBM licensed program that facilitates the process of installing and servicing an z/OS system.

system operations. The part of SA z/OS that monitors and controls system operations applications and subsystems such as NetView, SDSF, JES, RMF, TSO, RODM, ACF/VTAM, CICS, IMS, and OPC. Also known as SysOps.

system services control point (SSCP). In SNA, the focal point within an SNA network for managing the configuration, coordinating network operator and problem determination requests, and providing directory support and other session services for end users of the network. Multiple SSCPs, cooperating as peers, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its domain.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks.

System/390 microprocessor cluster. A configuration that consists of central processor complexes (CPCs) and may have one or more integrated coupling facilities.

T

TAF. Terminal access facility.

target. A processor or system monitored and controlled by a focal-point system.

target control task. In SA z/OS, target control tasks process commands and send data to target systems and workstations through communications tasks. A target control task (a NetView autotask) is assigned to a target system when the target system is initialized.

target hardware. In SA z/OS, the physical hardware on which a target system runs. It can be a single-image or physically partitioned processor. Contrast with *target system*.

target system. (1) In a distributed system environment, a system that is monitored and controlled by the focal-point system. Multiple target systems can be controlled by a single focal-point system. (2) In SA z/OS, a computer system attached to the focal-point system for monitoring and control. The definition of a target system includes how remote sessions are established, what hardware is used, and what operating system is used.

task. (1) A basic unit of work to be accomplished by a computer. (2) In the NetView environment, an operator station task (logged-on operator), automation operator (autotask), application task, or user task. A NetView task performs work in the NetView environment. All SA z/OS tasks are NetView tasks. See also *communications task*, *message monitor task*, and *target control task*.

telecommunication line. Any physical medium, such as a wire or microwave beam, that is used to transmit data.

terminal access facility (TAF). (1) A NetView function that allows you to log onto multiple applications either on your system or other systems. You can define TAF sessions in the SA z/OS customization panels so you don't have to set them up each time you want to use them. (2) In NetView, a facility that allows a network operator to control a number of subsystems. In a full-screen or operator control session, operators can control any combination of subsystems simultaneously.

terminal emulation. The capability of a microcomputer or personal computer to operate as if it were a particular type of terminal linked to a processing unit to access data.

threshold. A value that determines the point at which SA z/OS automation performs a predefined action. See *alert threshold*, *warning threshold*, and *error threshold*.

time of day (TOD). Typically refers to the time-of-day clock.

Time Sharing Option (TSO). An optional configuration of the operating system that provides conversational time sharing from remote stations. It is an interactive service on z/OS, MVS/ESA, and MVS/XA.

Time-Sharing Option/Extended (TSO/E). An option of z/OS that provides conversational timesharing from remote terminals. TSO/E allows a wide variety of users to perform many different kinds of tasks. It can handle short-running applications that use fewer sources as well as long-running applications that require large amounts of resources.

timers. A NetView command that issues a command or command processor (list of commands) at a specified time or time interval.

TOD. Time of day.

token ring. A network with a ring topology that passes tokens from one attaching device to another; for example, the IBM Token-Ring Network product.

TP. Transaction program.

transaction program. In the VTAM program, a program that performs services related to the processing of a transaction. One or more transaction programs may operate within a VTAM application program that is using the VTAM application program interface (API). In that situation, the transaction program would request services from the applications program using protocols defined by that application program. The application program, in turn, could request services from the VTAM program by issuing the APPCCMD macro instruction.

transitional automation. The actions involved in starting and stopping subsystems and applications that have been defined to SA z/OS. This can include issuing commands and responding to messages.

translating host. Role played by a host that turns a resource number into a token during a unification process.

trigger. Triggers, in combination with events and service periods, are used to control the starting and stopping of applications in a single system or a parallel sysplex.

TSO. Time Sharing Option.

TSO console. From this 3270-type console you are logged onto TSO or ISPF to use the runtime panels for I/O operations and SA z/OS customization panels.

TSO/E. TSO Extensions.

U

UCB. The unit control block; an MVS/ESA data area that represents a device and that is used for allocating devices and controlling I/O operations.

unsolicited message. An SA z/OS message that is not a direct response to a command. Contrast with *solicited message*.

user task. An application of the NetView program defined in a NetView TASK definition statement.

Using. An RMF Monitor III definition. Jobs getting service from hardware resources (processors or devices) are **using** these resources. The use of a resource by an address space can vary from 0% to 100% where 0% indicates no use during a Range period, and 100% indicates that the address space was found using the resource in every sample during that period. See also *Workflow*.

V

view. In the NetView Graphic Monitor Facility, a graphical picture of a network or part of a network. A view consists of nodes connected by links and may also include text and background lines. A view can be displayed, edited, and monitored for status information about network resources.

Virtual Storage Extended (VSE). An IBM licensed program whose full name is Virtual Storage Extended/Advanced Function. It is an operating system that controls the execution of programs.

Virtual Telecommunications Access Method (VTAM). An IBM licensed program that controls communication and the flow of data in an SNA network. It provides single-domain, multiple-domain, and interconnected network capability. Its full name is Advanced Communications Function for the Virtual Telecommunications Access Method. Synonymous with *ACF/VTAM*.

VM/ESA. Virtual Machine/Enterprise Systems Architecture.

VM Second Level Systems Support. With this function, Processor Operations is able to control VM second level systems (VM guest systems) in the same way that it controls systems running on real hardware.

volume. A direct access storage device (DASD) volume or a tape volume that serves a system in an SA z/OS enterprise.

volume entry. A construct, created with the customization dialogs, used to represent and contain policy for a volume.

volume group. A named set of volumes. A volume group is part of a system definition and is used for monitoring purposes.

volume group entry. An construct, created with the customization dialogs, used to represent and contain policy for a volume group.

Volume Workflow. The SA z/OS Volume Workflow variable is derived from the RMF Resource Workflow definition, and is used to measure the performance of volumes. SA z/OS calculates Volume Workflow using:

```
                     accumulated
                        Using
Volume     =  -------------------------  * 100
Workflow %    accumulated + accumulated
                 Using        Delay
```

The definition of **Using** is the percentage of time when a job has had a request accepted by a channel for the volume, but the request is not yet complete.

The definition of **Delay** is the delay that waiting jobs experience because of contention for the volume. See also *Address Space Workflow*.

VSE. Virtual Storage Extended.

VTAM. Virtual Telecommunications Access Method.

W

warning threshold. An application or volume service value that determines the level at which SA z/OS changes the associated icon in the graphical interface to the warning color. See *alert threshold*.

workflow. See *Address Space Workflow* and *Volume Workflow*.

workstation. In SA z/OS workstation means the *graphic workstation* that an operator uses for day-to-day operations.

write-to-operator (WTO). A request to send a message to an operator at the z/OS operator console. This request is made by an application and is handled by the WTO processor, which is part of the z/OS supervisor program.

write-to-operator-with-reply (WTOR). A request to send a message to an operator at the z/OS operator console that requires a response from the operator. This request is made by an application and is handled by the WTO processor, which is part of the z/OS supervisor program.

WTO. Write-to-Operator.

WTOR. Write-to-Operator-with-Reply.

WWV. The US National Institute of Standards and Technology (NIST) radio station that provides standard time information. A second station, known as WWVB, provides standard time information at a different frequency.

X

XCF. Cross-system coupling facility.

XCF couple data set. The name for the sysplex couple data set prior to MVS/ESA System Product Version 5 Release 1. See also *sysplex couple data set*.

XCF group. A set of related members that a multisystem application defines to XCF. A member is a specific function, or instance, of the application. A member resides on one system and can communicate with other members of the same group across the sysplex.

XRF. Extended recovery facility.

Numerics

390-CMOS. Processor family group designator used in the SA z/OS processor operations documentation and in the online help to identify any of the following S/390 CMOS processor machine types: 9672, 9674, 2003, 3000, or 7060. SA z/OS processor operations uses the OCF facility of these processors to perform operations management functions. See *OCF-based processor*.

Index

Readers' Comments — We'd Like to Hear from You

System Automation for z/OS
Defining Automation Policy
Version 3 Release 2

Publication No. SC33-8262-05

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Submit your comments using one of these channels:
- Send your comments to the address on the reverse side of this form.
- Send a fax to the following number: FAX (Germany): 07031 16-3456
  FAX (Other Countries): +49 7031 16-3456
- Send your comments via e-mail to: s390id@de.ibm.com

If you would like a response from IBM, please fill in the following information:

Name	Address
Company or Organization	
Phone No.	E-mail address

Readers' Comments — We'd Like to Hear from You
SC33-8262-05

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Deutschland Research & Development GmbH
Department 3248
Schoenaicher Strasse 220
71032 Boeblingen
Germany

Fold and Tape **Please do not staple** Fold and Tape

Cut or Fold
Along Line

Program Number: 5645-006

Printed in USA

SC33-8262-05