NetView for AIX Administrator's Guide Version 4

Document Number SC31-8168-01

April 26, 1996

NetView for AIX

SC31-8168-01

Administrator's Guide

Version 4

NetView for AIX

SC31-8168-01

Administrator's Guide

Version 4

Note

Before using this product, read the general information under “Notices” on page ix.

First Edition (July 1995)

This document applies to IBM NetView for AIX (feature 5608), which is a feature of SystemView for AIX (5765-527). IBM NetView for AIX runs under the AIX Operating System for RISC System/6000 Version 3 Release 2 (5756-030) or Version 4 Release 1 (5765-393). This product is based, in part, on Hewlett-Packard Company's OpenView product.

Publications are not stocked at the address given below. If you want more IBM publications, ask your IBM representative or write to the IBM branch office serving your locality.

A form for your comments is provided at the back of this document. If the form has been removed, you may address comments to:

IBM Corporation
Department CGMD
P.O. Box 12195
Research Triangle Park, North Carolina 27709
U.S.A.

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

Contents

Figures

Tables

Notices

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

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

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594
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:

Site Counsel
IBM Corporation
P.O. Box 12195
3039 Cornwallis Road
Research Triangle Park, NC 27709-2195
USA

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

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

Trademarks

The following terms, denoted by an asterisk (*) at their first occurrences in this publication, are trademarks of IBM Corporation in the United States or other countries:

AIX	OS/2	RISC System/6000
AIXwindows	NetView	SystemView
IBM	PS/2	

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

About This Book

The *NetView for AIX Administrator's Guide* provides information for operating the IBM* NetView* for AIX* (hereafter referred to as NetView for AIX) program. It explains what this program does and how to use it to manage and monitor a multiprotocol network.

Who Should Use This Book

The *NetView for AIX Administrator's Guide* is for system administrators or anyone who already has a basic familiarity with the NetView for AIX program. You should also have a basic understanding of networking and the AIX operating system. Most of the tasks in this book require a read-write map and root authority.

If you are not familiar with NetView for AIX, refer to *NetView for AIX User's Guide for Beginners*.

How to Use This Book

This book is organized as follows:

- Chapter 1, “Understanding NetView for AIX Processes” on page 1 describes the processes, applications, and databases used by NetView for AIX. This information was in the *NetView for AIX Programmer's Guide*. Read this chapter if you need an overview of NetView for AIX.
- Chapter 2, “Defining and Managing a Security Policy” on page 19 is intended for security administrators or anyone is who responsibile for managing NetView for AIX security. This chapter describes how to use NetView for AIX security services to control access to NetView for AIX and includes information about how to create user and group profiles, collect and view audit data, and distribute a central security policy to other servers in your network.
- Chapter 3, “Creating and Customizing Submaps” on page 43 provides information about creating a customized submap hierarchy, which includes creating and customizing maps, submaps, and objects. This chapter also includes information about how to manage maps in a distributed network environment (client/server). In addition, this chapter describes how to use the collection facility to group objects into a group (collection). Defining a collection creates a submap of objects that meet the definition criteria you specify.
- Chapter 4, “Customizing the Graphical Interface” on page 93 explains how to customize the graphical interface. Once your maps and submaps are created, use this information to customize the presentation of information. For example, you can arrange symbols, assign maps, and change the background.
- Chapter 5, “Correlating, Filtering, and Configuring Events” on page 109 presents information about creating event correlation rules and defining event filters to control the events that are displayed. This chapter also describes how to configure events.

- Chapter 6, “Managing Network Configuration” on page 157 describes how to manage network configuration using some of the tools and menu operations provided by the NetView for AIX program. This chapter contains information about configuring for manager backup.
- Chapter 7, “Managing Network Performance” on page 183 describes how to monitor network performance using some of the tools and menu options provided by the NetView for AIX program. This chapter contains information about monitoring the system resources on your management system.
- Chapter 8, “Using the Agent Policy Manager” on page 211 describes how to use the Agent Policy Manager to set up and view information about thresholds and file monitoring in a network. You must have the Systems Monitor Version 2 Mid-Level Manager or System Information Agent installed in your network to use the Agent Policy Manager.
- Appendix, “NetView for AIX Internal Traps” on page 241 lists the traps generated by NetView for AIX.

Highlighting and Operation Naming Conventions

The following highlighting conventions are used in this book, with the noted exceptions:

Bold	Identifies commands and shell script paths (except in reference information), default values, user selections, daemon paths (on first occurrence), and flags (in parameter lists).
Italics	Identifies parameters whose actual names or values are to be supplied by the user, and terms that are defined in the following text.
`Monospace`	Identifies subjects of examples, messages in text, examples of portions of program code, examples of text you might see displayed, information you should actually type, and examples used as teaching aids.

The NetView for AIX operation naming convention used in this book shows the location of the operation in relation to the menu bar or context menu. The naming convention follows the format shown in this example:

```
Monitor..Network Configuration..Addresses
```

In this example, `Monitor` is a menu bar or context menu option, `Network Configuration` is an operation available from the Monitor submenu, and `Addresses` is an option that is available when you select Network Configuration.

Some operations require you to make selections from several layers of submenus before you reach the submenu containing the operation.

Where to Find More Information

The “Bibliography” on page 283 describes publications that can be helpful when using the NetView for AIX program. The Internet Request for Comments (RFC) documents

listed are shipped on the NetView for AIX program installation media and are installed in the /usr/OV/doc directory.

The following sources provide specific information that is not documented in the IBM NetView for AIX Version 4 library:

- The /usr/lpp/nv6000/README file provides additional information about the NetView for AIX program.
- The online help facility provides task, dialog box, and graphical interface information to help you use this program.
- For more information about Simple Network Management Protocol (SNMP), Transmission Control Protocol/Internet Protocol (TCP/IP), and general network basics, the following list is recommended reading:

  Rose, Marshall T. *The Simple Book: An Introduction to Management of TCP/IP-based Internets*. Englewood Cliffs, NJ: Prentice-Hall, 1994 (ISBN 0-13-177254-6)

  Comer, Douglas. *Internetworking with TCP/IP: Principles, Protocols, and Architecture, Volume 1*. New York, NY: Prentice-Hall, 1991. (ISBN 0-13-468505-9)

  Black, Uyless. *Network Management Standards. The OSI, SNMP, and CMOL Protocols*. New York, NY: McGraw-Hill, 1992. (ISBN 0-07-005554-8)

Chapter 1. Understanding NetView for AIX Processes

The NetView for AIX program uses many processes and databases to perform network management functions. To administer your network effectively, you should be familiar with the basic operation and interactions among the parts of the NetView for AIX program.

This chapter describes each part of the NetView for AIX program and gives a summary of its operation. Read this chapter if you need to understand how the different parts of the NetView for AIX program work.

The NetView for AIX program uses two types of processes:

- Foreground processes, or applications, that run or can be invoked while the graphical interface is running.
- Background processes, or *daemons*, that run continuously regardless of whether the graphical interface is running. These processes can be started only by the root user or the root shell and stopped only by the root user. Generally, the daemons provide services that must be available at all times.

Foreground Processes

The processes, or applications, listed in Table 1 provide the interface through which you manage your network. Many of these processes correspond to selections you can make from the main menu.

Table 1. The NetView for AIX Foreground Processes

Process Category	Process Name
Principal NetView for AIX graphical interface	ovw
Map display	ipmap (IP topology) xxmap (non-IP topology)
Event display	nvevents
Menu operations	xnmtrap xnmsnmpconf xnmloadmib xnmloadmib2 xnmbrowser xnmbrowser2 xnmcollect xnmbuilder xnmfault xnmgraph xnmrunreport nmpolling backup shpmon
Dialog box management	xnmappmon

The ovw Application

The **/usr/OV/bin/ovw** application is the principal NetView for AIX graphical interface for managing TCP/IP-based internets. The ovw application provides map drawing, map editing, and menu management operations. The ovw application is an X11/Motif application based on OSF/Motif user interface guidelines.

To start the ovw application, execute the **/usr/OV/bin/nv6000** shell script. The nv6000 shell script supports the same options as those supported by the ovw application. The ovw application automatically starts the ipmap, xxmap, and nvevents applications.

If you are the root user, the **nv6000** shell script starts the daemon processes and the ovw application. If you are not the root user, the shell script starts only the ovw application.

Once the ovw application is operating, some daemons and applications are dynamically updated through events forwarded by the ovesmd daemon. Table 2 shows where the ovesmd daemon sends events, and how those events are used.

Table 2. Where the ovesmd Daemon Sends Events

Destination	Use
The netmon daemon	Tracks changes in the state of the network.
The ovtopmd daemon	Updates the topology database.
The ipmap application	Informs the graphical interface of changes in the IP topology database and informs ovtopmd of user-initiated changes to map databases.
The nvevents application	Displays the event in either the Event Cards or List format.
The tralertd daemon	Forwards events to the NetView program as alerts.

The ipmap Application

The **/usr/OV/bin/ipmap** application is started by the ovw application automatically. The ipmap application ensures that the ovw application (graphical interface) and the ovtopmd daemon behave consistently. For example, when an object is deleted using the graphical interface, the graphical interface tells ipmap which symbols and objects were removed. The ipmap application then tells ovtopmd to make the appropriate changes to the topology database.

In another example, when the netmon daemon discovers a new node, the ovtopmd daemon adds the node to the IP topology database and informs ipmap that a new node has been discovered. The ipmap application uses what it knows about IP devices to tell the graphical interface which icon and connection symbols it needs to create. The graphical interface then displays the correct symbols and modifies the map database accordingly.

When the ipmap application is started, it queries the ovw application to determine when the map was last open. Next, it calls the ovtopmd daemon to find out all changes to the IP topology database since the last time the map was open using the graphical interface. The ipmap application determines what, if anything, has changed since the map was last open, and then tells the ovw application to add, change, or delete the appropriate icon and connection symbols. This process is called synchronization.

If the ipmap application can find an association between an SNMP node's sysObjectID MIB variable and a symbol type in the oid_to_sym file, the graphical interface displays the node by drawing the appropriate symbol on the submap. If the oid_to_sym file does not contain a matching sysObjectID entry, the ipmap application extracts attributes from the topology database and tells the ovw application to create and display a generic symbol.

Once the synchronization phase is completed, the ipmap application is updated based on information received dynamically from the ovtopmd daemon, which receives updates from the netmon daemon. The netmon daemon continuously monitors the state of your network by sending SNMP requests to SNMP-managed nodes and ICMP requests to non-SNMP IP nodes. The netmon daemon communicates changes to the ovtopmd

daemon, which updates the topology database and informs the ovwdb daemon that the object database needs to be updated. If the change affects the IP Map, the ipmap application notifies the ovw application. Otherwise, the change is reflected only in the object database and the IP topology database.

The xxmap Application

The **/usr/OV/bin/xxmap** application is very similar to the ipmap application with one important difference. The xxmap application processes non-IP topology information that is stored by the gtmd daemon in the general topology database. The ipmap application processes IP information that is stored by the ovtopmd daemon in the IP topology database. The non-IP topology information stored in the general topology database comes from non-IP discovery applications or agents that use the general topology MIB format to send topology information to the gtmd daemon.

The xxmap application ensures that the maps being displayed are synchronized with the contents of both the general topology and object databases. Using the /usr/OV/conf/oid_to_protocol file, the xxmap application matches the oid contained in the MIB to the correct submaps and symbols required to display the topology of a particular protocol. For more information about the xxmap application, see the *NetView for AIX Programmer's Guide*.

The nvevents Application

The **/usr/OV/bin/nvevents** application displays events in the main window Control Desk in either the Event Cards or List presentation format. When this process is invoked by the ovw application, it reads the ovevent.log file to recover events that occurred since the nvevents application was last running.

Following startup, nvevents receives SNMP traps that are filtered by the ovesmd daemon and uses those traps to update the event cards or list.

It also monitors the status of the snmpCollect, netmon, or ovtopmd daemons. If one of those daemons is not running, nvevents presents a warning box to notify you.

The xnmtrap Application

The **/usr/OV/bin/xnmtrap** application is invoked by the Options..Event Configuration..Trap Customization: SNMP... menu option. This option helps you control how enterprise-specific events (traps) are handled. For example, you can customize the message displayed through nvevents when a particular event arrives. You can also specify a command or a script that should be executed when a particular event arrives. Event configuration changes are stored in the /usr/OV/conf/C/trapd.conf configuration file.

The xnmsnmpconf Application

The **/usr/OV/bin/xnmsnmpconf** application is invoked by the Options..SNMP Configuration menu option, which enables you to specify netmon status polling intervals, time-out intervals, number of retries, proxy information, and agent community names.

The SNMP configuration values are stored in the **/usr/OV/conf/ovsnmp.conf_db** file. Processes on the management station, such as the netmon daemon, MIB applications, and SNMP API-based applications, look up an agent's community name and other SNMP options in the **ovsnmp.conf_db** file when access to agent MIB values through SNMP requests is required.

The xnmloadmib Application

The **/usr/OV/bin/xnmloadmib** application is invoked by the Options..Load/Unload MIBs: SNMP... menu option. Select this option when you want to load new SNMPv1 Internet-standard, enterprise-specific, or general topology MIBs into the MIB database.

Once you have loaded the new MIB into the MIB database, you can use the MIB Browser, MIB Data Collection, and MIB Application Builder options, as well as the applications built by the MIB application builder, to manage your multivendor network.

The xnmloadmib2 Application

The **/usr/OV/bin/xnmloadmib2** application is invoked by the Options..Load/Unload MIBs: SNMP..SNMPv1/SNMPv2 menu option. Select this option when you want to load new SNMPv1 or SNMPv2 Internet-standard, enterprise-specific, or general topology MIBs into the MIB database.

Once you have loaded the new MIB into the MIB database, you can use the MIB Browser, MIB Data Collection, and MIB Application Builder options, as well as the applications built by the MIB application builder, to manage your multivendor network.

The xnmbrowser Application

The **/usr/OV/bin/xnmbrowser** application is invoked by the Tools..MIB Browser: SNMP menu option, which enables you to get and set MIB values for Internet-standard and enterprise-specific MIB objects on SNMPv1 agents. This application hides the actual SNMP requests used to perform these operations, so you need to know only which MIB objects you want to access, not the commands required to do so.

The xnmbrowser2 Application

The **/usr/OV/bin/xnmbrowser2** application is invoked by the Tools..MIB Browser: SNMPv1/SNMPv2 menu option, which enables you to get and set MIB values for Internet-standard and enterprise-specific MIB objects on SNMPv1, SNMPv2C, and secure SNMPv2USEC agents. This application hides the actual SNMP requests used to perform these operations, so you need to know only which MIB objects you want to access, not the commands required to do so.

The xnmcollect Application

The **/usr/OV/bin/xnmcollect** application is invoked by the Tools..Data Collection and Thresholds: SNMP menu option, which enables you to configure the manager to collect MIB data from network objects at regular intervals. The configuration information is stored in the /usr/OV/conf/snmpCol.conf configuration file. The collected data is stored in files in the /usr/OV/databases/snmpCollect directory.

Once you have stored the collected data, you can use the xnmgraph application to view the collected data or import the data into your own application using the **snmpColDump** command. You can also define thresholds for the collected MIB data and generate events when the specified thresholds are exceeded.

The xnmbuilder Application

The **/usr/OV/bin/xnmbuilder** application is invoked by the Tools..MIB Application Builder: SNMP menu option, which enables you to build custom screens to manage multivendor MIB objects. The information you define using the MIB application builder is stored in registration files and help files. The xnmgraph application displays **Graph** applications, while the xnmappmon application displays **Form** and **Table** applications.

The xnmgraph Application

The **/usr/OV/bin/xnmgraph** application enables you to graph the results of monitoring operations for managed SNMP objects selected from the map. The results may be real time or collected historical data.

When you select the Tools..Graph Collected Data: SNMP option on the main menu, the ovw application forwards the objects previously selected on the map to the xnmgraph application. The xnmgraph application queries the selected objects and displays the results in a line graph.

The xnmfault Application

The **/usr/OV/bin/xnmfault** application enables you to locate, for any node on the map, all of its component resources that have failed.

When you select a node on a submap, and then select Tools..Locate Failing Resources option on the main menu, the xnmfault application creates a submap showing all the resources that are children of the selected node and that have failed. Be sure to close this submap through the navigation tree. If not, the xnmfault application is not notified that the submap has been closed. The submap remains open and can be accessed from the Navigation Tree.

The xnmrunreport Application

The **/usr/OV/bin/xnmrunreport** application corresponds to the Monitor..Reports: Site Provided menu option. You can use this application to generate reports that send output directly to the screen, which is useful for problem determination or real-time network monitoring. You can also generate reports that store information in flat files or databases for future reference.

The nmpolling Application

The **/usr/OV/bin/nmpolling** application is accessed through the Options..Topology/Status Polling Intervals: IP menu option that enables you to configure certain netmon polling intervals.

The backup Application

The **/usr/OV/bin/backup** application is accessed through the Administer..Backup menu option. This process lets you configure managers to backup another manager's objects if the other manager should go down. You configure objects as manager nodes or managed containers and then specify managers as an active manager or a backup manager.

The shpmon Application

The **/usr/OV/bin/shpmon** application is accessed through the Monitor..Local Filesystem & Paging Space menu option. Using this option, you can receive a trap or message letting you know when a threshold condition has reached its limit. The shpmon application monitors the root filesystem and the paging space on the local manager system where NetView for AIX is installed.

The xnmappmon Application

The **/usr/OV/bin/xnmappmon** application manages dialog boxes that contain the text output of monitoring operations performed on managed SNMP objects that have been selected from the map. This application is also called an application encapsulator.

When you select a monitoring operation from the NetView for AIX main menu bar, the ovw application forwards the objects selected on the map to the xnmappmon application as input. The xnmappmon application displays the appropriate dialog box, translates the selected operation and object or objects into a command, and executes the command.

When processing is completed, the command returns its text output to the xnmappmon application for display in the dialog box. In the case of the Locate..Route menu option, xnmappmon also returns to ovw a list of map objects to highlight as output of the network management operation.

For example, the xnmappmon application can be used to execute the **mibform, mibtable, rnetstat, findroute, rbdf**, and **rping** commands. These commands help you monitor and diagnose problems in your TCP/IP network.

Process Management

As a network administrator, you will use several process management commands to control the operation of NetView for AIX daemons. These commands are:

ovstart Starts all daemons or selected daemons, depending on the options you specify.

ovstop Stops all daemons or selected daemons, depending on the options you specify.

ovstatus Shows the status of all daemons or selected daemons, depending on the options you specify, including the process management daemon itself.

nvstatus Shows the status of all daemons or selected daemons, depending on the options you specify, that are running on the server. This command can be run only on a client workstation.

The process management daemon, ovspmd, coordinates the starting and stopping of daemons that communicate with the NetView for AIX program.

The **ovstart** command starts the ovspmd daemon, which in turn starts the other NetView for AIX daemons in a particular order. The information about startup order is in the startup configuration file, /usr/OV/conf/ovsuf.

The startup configuration file is constructed by the **ovaddobj** command. The ovaddobj process takes information from the local registration files (LRF) and places it in the startup configuration file.

The ovspmd daemon receives requests from ovstart command and sends status responses to ovstart. It starts all NetView for AIX daemons (if they are listed in the ovsuf file) and maintains a communication channel with each of them. These daemons should always be running. The ovspmd daemon starts the host connection daemons, spappld and tralertd, only if the AIX Service Point is installed. A check for the AIX Service Point is performed during installation of NetView for AIX.

Background Processes

The daemons fall into the following categories:

- Topology discovery and database operation
- Event and trap processing
- Security services
- Host connection

Topology Discovery and Database Daemons

The NetView for AIX program discovers and updates the topology of IP networks and translates the information into symbols that appear on the views you see of your network's map. The program also facilitates the discovery of networks that use non-IP protocols for communication, so you can extend your network management coverage.

Figure 1 illustrates the interactions among the topology discovery and database daemons. Each daemon is described in the following sections.

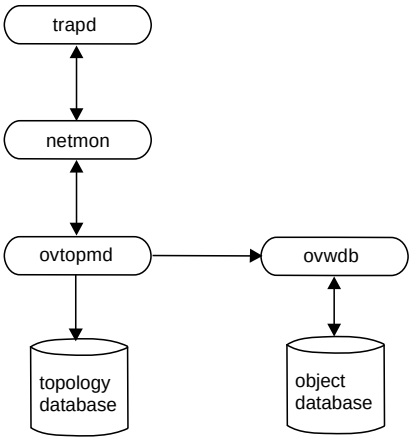


Figure 1. Interactions among Topology Discovery and Database Daemons

The mgragentd Daemon

The **/usr/OV/bin/mgragentd** daemon runs on the manager station to determine the status of the NetView for AIX program and to respond to queries from other manager stations. In a distributed network environment, the mgragentd daemon is used to determine the status of the NetView for AIX daemons. This daemon requires no user configuration or manipulation.

The netmon Daemon

The **/usr/OV/bin/netmon** daemon polls SNMP agents to discover network topology. During initial discovery, the netmon daemon sends an SNMP trap to the trapd daemon to inform it of newly discovered network entities. Some of these entities may be agents that communicate with non-IP networks or devices. As long as the entities have an IP address, the netmon daemon will discover them.

After initial discovery is complete, the netmon daemon polls the SNMP agents to detect topology, configuration, and status changes in the IP network, and sends corresponding traps to the trapd daemon. However, the netmon daemon does not have a connection with non-IP networks or devices. Topology changes for these network entities must be communicated to the NetView for AIX program in other ways. See “The noniptopod Daemon” on page 11 and “The gtmd Daemon” on page 11 for more information about non-IP topology discovery.

In addition to polling SNMP agents, the netmon daemon polls network nodes. To check an SNMP-managed node's MIB values, the netmon daemon uses an SNMP request. To check the status of all nodes, the netmon daemon uses ICMP echo requests (ping). Based on the discovered information, the netmon daemon generates and updates the topology map.

The netmon polling values and information about network objects, including their relationships, status, and thresholds, are stored in a set of files called the *topology*

database. If the database does not initially exist, the ovtopmd daemon creates it during discovery of the network's topology and automatic generation of the map. When the user starts up the graphical interface, the ipmap application compares the contents of the topology database with the contents of the graphical interface's map databases. The ipmap application tells the graphical interface what has changed since its last invocation, and then the graphical interface updates the map.

The netmon daemon assumes that your initial network management region is composed of the network or networks to which the NetView for AIX program, the management station, is directly connected. If you want a different initial configuration, you can provide a seed file, which contains a list of nodes you want to appear on the automatically-generated network map. See “Using a Seed File to Control Network Discovery” on page 159 for information about seed files.

The nvlockd Daemon

The **/usr/OV/bin/nvlockd** helps the gtmd daemon and xxmap application control access to the ovwdb daemon. This daemon requires no user configuration or manipulation.

The ovtopmd Daemon

The **/usr/OV/bin/ovtopmd** daemon maintains the network topology database. The topology database is a set of files in the /usr/OV/databases/openview/topo directory that store netmon polling values and other information about network objects, including their relationships and status.

The ovtopmd daemon generates and updates the topology database using status information obtained from the netmon daemon. The ovtopmd daemon also checks for existing non-IP objects with which to correlate.

The ovwdb Daemon

The **/usr/OV/bin/ovwdb** daemon controls the NetView for AIX object database. This database stores object information that the graphical interface uses to generate output for Describe operations. For example, when you select the Edit..Modify/Describe..Objects option, you can view several attributes for the objects you have selected. The information you see is retrieved from the NetView for AIX object database.

If the netmon daemon detects a change in the network, it calls the ovtopmd daemon to update the topology database. In turn, the ovtopmd daemon calls the ovwdb daemon to update the NetView for AIX object database.

In order for the ovw application to run, the ovwdb daemon must first be running. The object database must be accessible to the ovw application so that the default submap can be generated.

The noniptopod Daemon

Initial topology discovery is handled by the netmon daemon, which sends an SNMP trap to the trapd daemon when a new network object is discovered. The trapd daemon then forwards traps, via the ovesmd daemon, to the **/usr/OV/bin/noniptopod** daemon, which accepts any trap that meets the following criteria:

- The discovered object has an IP address.
- The discovered object could be an agent that can communicate with a non-IP device or network.

The noniptopod daemon sends an SNMP get command for each OID listed in the /usr/OV/conf/oid_to_command file to the new network object associated with the trap. It checks to see if the OIDs are supported by the agent. If a non-null response is sent back, the commands associated with the supported OIDs are started.

The command in the oid_to_command file activates the non-IP protocol's proprietary daemon, which sends a request to the agent to gather all non-IP topology information and forward it to the proprietary daemon for conversion to the general topology MIB format. The MIB information is then sent to the gtmd daemon.

The gtmd Daemon

The **/usr/OV/bin/gtmd** daemon receives information sent by non-IP discovery applications, agents, and proxy agents that use the general topology MIB format to describe the attributes of devices on a non-IP network. Information can be received in traps or in API calls. This daemon stores the non-IP topology information in its own database and correlates it with IP topology information stored in the object database to determine whether an IP object can also be identified as having an association with a non-IP protocol. The gtmd daemon makes the topology information available for display through the xxmap application.

Non-IP discovery applications can register with the gtmd daemon to receive notifications of changes to the topology information or to receive topology data. The gtmd daemon updates its database each time a trap is received and notifies registered applications of the operation performed. The gtmd and noniptopod daemons are, by default, not started when the NetView for AIX program is started. You can use SMIT to configure them and indicate that they should be started by the **nv6000** command.

For more information about the NetView for AIX general topology function and the gtm API, refer to the *NetView for AIX Programmer's Guide*.

The nvotd Daemon

The **/usr/OV/bin/nvotd** daemon receives non-IP topology events from the gtmd daemon and forwards them to the Event Display application (nvevents) if API calls are used to send information to the gtmd daemon. The nvotd daemon, by default, is not started when the NetView for AIX program is started. You can use SMIT to configure the nvotd daemon and indicate that the daemon should be started by the **nv6000** command.

The C5d Daemon

The **/usr/OV/bin/C5d** daemon coordinates setting threshold monitor and file monitor definitions for Systems Monitor Mid-Level Managers (MLMs) and Systems Information Agents (SIAs). The C5d daemon is, by default, not started when the NetView for AIX program is started. You can use SMIT to configure the C5d daemon so that it starts when the **nv6000** command is used.

The nvcold Daemon

The **/usr/OV/bin/nvcold** daemon maintains collections of objects as they have been defined by users or applications.

Event and Trap Processing Daemons

Events and traps provide information about changes in the status of network elements and alert the NetView for AIX program to occurrences in the network. When events and traps are received, they must be routed to the appropriate applications and logged for future reference.

The daemons that perform these tasks can be divided into the following three groups:

- Communications infrastructure daemons, including the pmd daemon and the orsd daemon, that control all CMOT and some SNMP communications entering and leaving the NetView for AIX program
- Event management services daemons, including the ovesmd daemon and the ovelmd daemon, which route and log all events received by the NetView for AIX program
- Other event and trap processing daemons, including the trapd daemon, the nvcorrd daemon, the actionsvr daemon, the nvpagerd daemon, the nvserverd daemon, the trapgend daemon, and the snmpCollect daemon

Figure 2 illustrates the interactions among the event and trap processing daemons. Each daemon is described in the following sections.

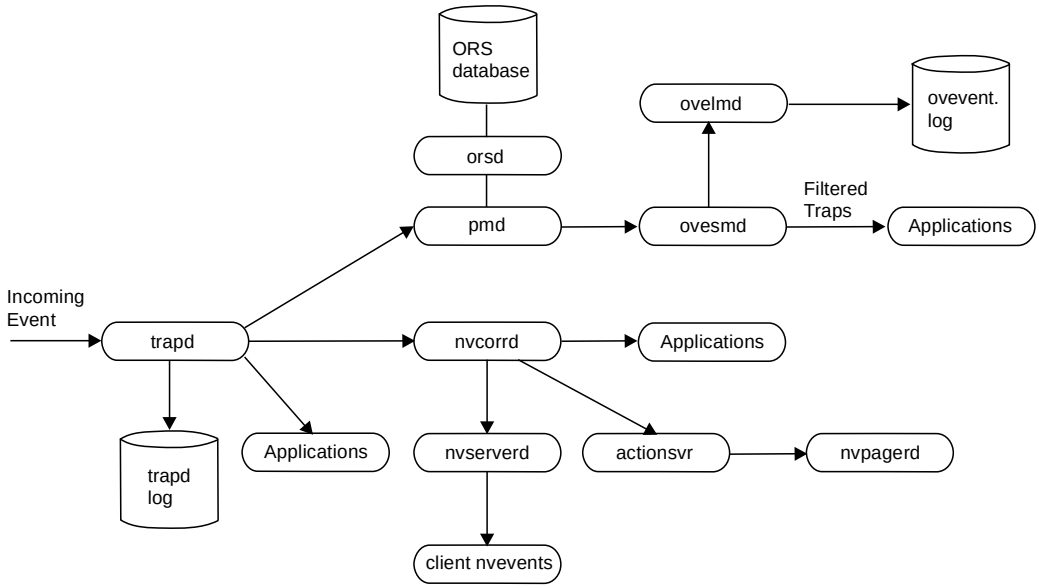


Figure 2. Interactions among Event and Trap Processing Daemons

The pmd Daemon

When an agent sends an event to an application, the event contains no routing information. It's similar to sending a letter without an address. The **/usr/OV/bin/pmd** daemon receives the events from the trapd daemon and forwards them to the ovesmd daemon. The ovesmd daemon forwards the events to the applications that have registered to receive them.

The pmd daemon centralizes the external communications for all applications and processes that use the CMOT protocol, as well as some of those using SNMP. This daemon contains the SNMP and CMOT protocol stacks.

The pmd daemon has two components:

- The locator function, which routes outbound requests to the appropriate agent. The locator function consults the data in the object registration database, which includes agent locations and the protocol used to access them. Note that application programs do not access the object registration database directly; instead, the pmd daemon automatically does so on their behalf.
- The association management function, which provides a way of sharing connections between network management nodes. This function enables application managers to share connections without having to bring up multiple connections between the nodes.

The orsd Daemon

The **/usr/OV/bin/orsd** daemon maintains the consistency of the CMIS object information in the object registration database. The object registration database is consulted by the pmd daemon to determine where an agent resides and which protocol to use to communicate with it.

The ovesmd Daemon

The **/usr/OV/bin/ovesmd** daemon is part of the NetView for AIX program's event management services. Known as the event sieve agent (ESA), the ovesmd daemon distributes events throughout the network based on the filters in effect for a particular application or user.

When events are sent from agents, they do not contain any routing information that the pmd daemon can use to forward them to the correct application. When the pmd daemon receives an event from trapd, it sends the event to the ovesmd daemon, which filters the event and forwards it to applications that have registered to receive it. The ovesmd daemon also forwards the event to the ovelmd daemon, which controls the event log.

The ovelmd Daemon

The **/usr/OV/bin/ovelmd** daemon is the event log agent (ELA), which stores SNMP traps, CMIS events, and event log configuration values in the /usr/OV/log/ovevent.log file. The ovevent.log and the ovevent.BAK files are binary files that are the source of information for the dynamic and historical event displays. It logs all SNMP traps received from the network and internal processes.

The trapd Daemon

The **usr/OV/bin/trapd** daemon receives traps from agents and internal processes and forwards them to the netmon, tralertd, nvcorrd, and the pmd daemons, and to the ipmap application. In return, the netmon daemon sends events to trapd when its polling reveals a change in the status of a network element.

The trapd daemon also forwards network events to other applications that have connected to it through the OVsnmp API.

The trapd daemon logs all received traps in the /usr/OV/log/trapd.log file. You can use any editor to look at the contents of the trapd.log file. You can also choose to turn trapd logging off.

The nvcorrd Daemon

The **/usr/OV/bin/nvcorrd** daemon receives events from the trapd daemon, correlates or compares the events to event processing decision and actions defined in rulesets registered with nvcorrd, and forwards them to registered applications, one of which is the Event Display application. The nvcorrd daemon passes events to the actionsvr daemon to manage actions defined in event correlation rulesets.

The actionsvr Daemon

When an action is to be processed in an event correlation rule, the nvcorrd daemon passes the action to the **/usr/OV/bin/actionsvr** daemon. The actionsvr daemon manages the action, starting a child process, while the nvcorrd daemon continues to process the event correlation ruleset. All actions requested and the events which caused those actions are logged in the /usr/OV/log/nvaction.alog and /usr/OV/log/nvaction.blog files.

The nvpagerd Daemon

The **/usr/OV/bin/nvpagerd** daemon manages the routing of the page command that is issued from the command line or within an event correlation rule.

When the paging action is to be processed in an event correlation rule, the nvcorrd daemon passes the action to the actionsvr daemon, which passes the action to the nvpagerd daemon. All paging actions requested and the events which caused those actions are logged in the /usr/OV/log/pagerd.log file.

The nvserverd Daemon

The **/usr/OV/bin/nvserverd** daemon receives events from the nvcorrd daemon and forwards them to different Event Display applications running on client workstations. The nvserverd daemon enables a user to select a set of events in one client application and work with it in all registered applications, clearing these events or changing the status of the events.

The trapgend Daemon

The **/usr/OV/bin/trapgend** daemon is a subagent (SMUX peer) provided with the NetView for AIX program that converts AIX alertable errors to SNMP traps.

On RISC System/6000 processors running AIX Version 3 Release 2 or later, system errors are logged by the operating system's error logging facilities in the /dev/error special file. An object installed by NetView for AIX in each system's object data manager (ODM) directs the AIX error logging daemon (errdemon) to inform the trap-notify process when alertable errors are logged. These alertable errors are forwarded by the trap-notify process to the trapgend daemon, which converts these alertable errors to SNMP traps.

Using the SNMP multiplexer (SMUX) protocol, trapgend forwards the traps to the AIX SNMP daemon, snmpd. The snmpd daemon then forwards the traps to the trapd daemon on the NetView for AIX manager specified by the trap destination.

The trapgend daemon also provides a trap throttle to suppress identical trap generation, enables remote ping operations from the NetView for AIX program, and supports CPU utilization and disk space monitoring MIB extensions.

The trapgend daemon must be installed on all nodes running AIX Version 3 Release 2 or later in your network if you want to receive information about CPU and disk space utilization.

The snmpCollect Daemon

The **/usr/OV/bin/snmpCollect** daemon collects, compares, and stores SNMP agent MIB values. It also checks the collected values against user-defined thresholds and generates events if the thresholds are exceeded.

Security Services Daemons

The security services daemons determine whether security is on or off, manage authentication and identification of NetView for AIX users, and manage communication between the security server and client workstations. Each daemon is described in the following sections.

The nvsecd Daemon

The **/usr/OV/bin/nvsecd** daemon determines if NetView for AIX security is on or off. If security is on, the nvsecd daemon requires each NetView for AIX user to login using a valid NetView for AIX user ID, password, and group ID. The nvsecd daemon checks the user's profile to verify login and checks the permissions defined in the user's group profile to control access to NetView for AIX resources. After login verification, the nvsecd daemon establishes a shared-key security context for each NetView for AIX process. The nvsecd daemon also monitors and stores security audit data.

The nvsecltd Daemon

The **/usr/OV/bin/nvsecltd** runs on client workstations and listens for communication from the security server. The nvsecltd daemon starts only when security is turned on and when at least one user logs into NetView for AIX on a client workstation. The nvsecltd daemon stops when the last client user logs out of the NetView for AIX program.

Host Connection Daemons

If your NetView for AIX program is connected to the NetView* program by an AIX Service Point program, you are using the services of the host connection daemons. These daemons facilitate both the conversion of SNMP traps to SNA alerts as well as all communication between the NetView for AIX program and the NetView program through the Service Point application. Each daemon is described in the following sections. For more information about the host connection, refer to *NetView for AIX and the Host Connection*.

The tralertd Daemon

The **/usr/OV/bin/tralertd** daemon is used in an environment where both TCP/IP and SNA protocols are running. The tralertd daemon receives events and traps generated or received by the NetView for AIX program. If a trap is so configured, the tralertd daemon converts it to an SNA alert, and sends the SNA alert to the NetView for 390 program through the AIX NetView Service Point program. If all the converted trap information cannot fit into the SNA alert, the original trap information is saved in the **/usr/OV/databases/tralertd** database and assigned a corresponding Log ID. The

NetView for 390 program uses this Log ID to query the tralertd database to retrieve the rest of the information.

The spappld Daemon

The **/usr/OV/bin/spappld** daemon provides a command interface between the NetView for 390 program on the host and the NetView for AIX program in an environment running SNA and TCP/IP protocols. The spappld daemon receives NetView RUNCMDs and executes their contents in the internet environment, and sends responses to the NetView program through the AIX NetView Service Point program.

Databases

This section describes the following databases:

- Map
- Object
- IP topology
- General
- Object registration service

These databases are controlled by the NetView for AIX processes; you cannot edit them directly.

The Map Database

The map database contains presentation information that is specific to each map. There is one map database per map. Examples of presentation information stored in the map database include the exact symbol placement on a map, the symbol associated with each object, and symbol labels. The ovw application updates the map database based on requests from the user or from other processes, such as the ipmap application. Use the **ovmapdump** command to view the contents of the map database. The map database is maintained by the ovw application.

The Object Database

The object database contains global object information. The information is generic; that is, it is not customized to any specific application. The object database contains information related to fields such as sysObjectID, vendor, and SNMP agent. When you choose the Edit..Modify/Describe..Object menu option, the information you see in the fields comes from this database. If netmon detects a status change in the network, netmon calls ovtopmd, which calls ovwdb to update the object database. Use the **ovobjprint** command to view the object database. The object database is maintained by the ovwdb daemon.

The IP Topology Database

The IP topology database contains topology information used during IP discovery and layout. The information in the topology database spans all maps. Much of the information in the NetView for AIX object database is duplicated in the topology database. Information in the topology database that is not in the object database includes state information for the netmon daemon.

The most important netmon state information includes time stamps that indicate when the object last changed and should next be polled. This information helps the netmon daemon detect changes so it can communicate the changes to the ovtopmd and trapd daemons. The topology database is controlled by the ovtopmd daemon and is updated based on information received from the netmon daemon. Use the **ovtopodump** command to view the contents of the IP topology database. The IP topology database is maintained by the ovtopmd daemon.

The General Topology Database

The general topology database stores topology information sent to the gtmd daemon. This database also stores information about submap grouping and content that has been defined by the protocol discovery applications or agents.

The xxmap application queries both this database and the NetView for AIX object database for display and semantic information. Any topology information stored in the general topology database can be deleted only by the agent that originally added the information. You cannot view the general topology database.

For more information about the use of the general topology database, refer to *NetView for AIX Programmer's Guide*.

The Object Registration Service Database

The object registration service (ORS) database contains location and protocol information for agents that use a protocol other than SNMP. This information helps to provide location transparency, which allows managers and agents to access objects and agents without using hard-coded addresses.

Chapter 2. Defining and Managing a Security Policy

You can use NetView for AIX security services to define a security policy for your network. This chapter, which is intended for security administrators or whoever is responsible for managing NetView for AIX security, will help you understand NetView for AIX security services so you can define a security policy that best suits your needs. This chapter also describes what you need to do to define a security policy. You will need to perform these tasks before turning security on. You might also need to perform some of these tasks after turning security on to add new NetView for AIX users or applications. In addition, this chapter describes the tasks to be performed in order to manage your security policy, such as, how to distribute the security configuration to other servers in your network and how to view audit data.

The following topics are described:

Understanding NetView for AIX Security Services

The key to effective security is understanding how the security features work and then enforcing the features. NetView for AIX security services provides the following controls:

- Network authentication and identification
- Protected network communication
- Password protection
- Continuous, auditable network management
- Network access control of NetView for AIX resources
- Customized NetView for AIX graphical interface
- Audit management
- Consistent security controls
- Pager service for event correlation

Network Authentication and Identification

Security services authenticates each user, allowing access to NetView for AIX when the user logs in using a valid NetView for AIX user ID, group ID, and password. Additional login controls include restricting user access to specific days and times and to specific client and server machines. You create a profile for each user that contains this infor-

mation. See “Managing NetView for AIX User Profiles” on page 27 for more detailed information.

Protected Network Communication

Communication between session partners (users, clients, servers) use a security context to ensure message integrity and identification. When the NetView for AIX server receives a request for authentication, the server verifies the log in and stores a security credential or ticket. This ticket allows shared-key security for each NetView for AIX process when establishing a session. Shared keys are used to validate messages between session partners by checking incoming messages to ensure they are from the correct sender.

The Log In Process

Users can log into NetView for AIX using one of the following methods:

- Log in using the NetView Authentication Dialog by entering the **nvauth** command on the AIX command line. The NetView Authentication Dialog is displayed, which contains input fields for the user's login ID, group ID, and password.
- Log in from the AIX command line using the **nvauth -login nvid nvgid** command.

  See the man page for more detailed information about the **nvauth** command.
- Start NetView for AIX. If a user starts NetView for AIX without first logging in, the NetView Authentication Dialog is displayed. NetView for AIX is initialized after successfully logging in.
- Use the Options..Shift_in operation on the NetView Authentication dialog box menu bar. See “Continuous, Auditable Network Management” on page 22 for a description of the Shift_in and Shift_out operations.

Users can log out using one of the following methods:

- Select the Tools..User Security operation on the NetView for AIX main menu bar to display the NetView Authentication dialog box. Then select the Options..Logout operation on the NetView Authentication dialog box menu bar.
- Enter the **nvauth -logout nvid** command on the AIX command line.
- Exit NetView for AIX. When a user exits NetView for AIX, the user is automatically logged out.

Logging out stops all user-initiated NetView for AIX processes. Refer to the *NetView for AIX User's Guide for Beginners* for more detailed information about login and logout.

Log In Considerations

Keep these things in mind about the NetView for AIX login process:

- Only one NetView for AIX login from the same AIX login ID is permitted. Subsequent NetView for AIX users from the same AIX login ID can start NetView for AIX without having to login. The first user ID is the process owner for audit purposes.

- If you have a security product installed, such as DCE, you can customize your security policy so that a NetView for AIX password is not required. Your security product manages the login procedure, allowing users to log in using a single password.

  By default, NetView for AIX requires a password unless you change this setting when you define the global security settings. See “Defining the Global Security Settings” on page 34 for more detailed information.

- A user issuing any server-restricted operation from a client workstation does not have to be logged in as a root user and will not be prompted for the root password. The user must, however, have the appropriate security permissions and the same AIX user ID on the server as on the client. Server-restricted operations are those operations that involve changing configuration files that exist on the server. These operations include:
  - Event configuration (Options..Event Configuration..Trap Customization:SNMP menu option or the **xnmtrap** command)
  - SNMP configuration (Options..SNMP Configuration menu option or the **xnmsnmpconf** command)
  - MIB data collection (Tools..Data Collection and Thresholds: SNMP menu option or the **xnmcollect** command)
  - Loading and unloading MIBs (Options..Load/Unload MIBs: SNMP menu option or the **xnmloadmib** command
  - Polling interval changes (Options..Topology/Status Polling Intervals: IP menu option or the **nmpolling** command)
  - Converting events to alerts (Options..Event Configuration..Trap to Alert Filter Control: SNMP menu option or the **tralertdfc** command)
  - Ruleset editor (Tools..Ruleset Editor menu option or the **nvrsEdit** command)
  - Security administration (Administer..Security Administration menu option or the **nvsec_admin** command)

Password Protection

The password supplied when a user tries to log in is encoded and compared with the encoded password stored in the security database. If the two match, the user gains access to NetView for AIX. Passwords are never stored or sent over the network in human-readable format.

You set the user's initial password, and you can change the user's password at any time, when you create or change the user's profile. See “Creating and Changing a User Profile” on page 30 for more detailed information.

The user can also change the password at any time by selecting the Operations..Change Password operation from the NetView Authentication dialog menu bar. See the *NetView for AIX User's Guide for Beginners* for more detailed information.

Continuous, Auditable Network Management

You might require that NetView for AIX manage your network without interruption. Operators using the same physical display can use the Shift_in and Shift_out operations to accomplish uninterrupted network monitoring. To use the Shift_in and Shift_out operations, operators must be in the same NetView for AIX group and use the same AIX login ID.

After the first operator logs in, the operator uses the Shift_out operation at the end of the shift. The Shift_out operation activates a window lock, allowing NetView for AIX processes to continue. The session is protected because only an authorized operator can remove the window lock. The next shift operator removes the window lock by clicking on the key displayed on the window lock screen to remove the window lock and then using the Operations.. Shift_in operation from the NetView Authentication Dialog box menu bar. Ownership of currently running NetView for AIX processes changes to the new operator for audit and identification purposes without having to restart each process.

The operator can regain the screen saver before using the Shift_in operation by selecting the **Reset** button on the NetView Authentication Dialog box.

Refer to the *NetView for AIX User's Guide for Beginners* for more detailed information about how to use the Shift_in and Shift_out operations.

Network Access Control of NetView for AIX Resources

User permissions or rights to access NetView for AIX resources are defined based on the NetView for AIX group to which the user belongs. Each NetView for AIX application provides a security registration file (SRF), which lists the application's resources, such as menu items, commands, and tools. The SRF includes the valid permissions (read, write, execute, and so on) for the application's resources.

Vendor applications can also register with security services to make use of and integrate with security services. The *NetView for AIX Programmer's Guide* provides detailed information about integrating an application with NetView for AIX's security application.

Each NetView for AIX group is associated with a security registration files list. The security registration files list is a collection of all the NetView for AIX applications that a group of users can access and the group's access permissions for all the elements within each application.

NetView for AIX provides the following pre-configured groups:

Oper
: Intended for network operators. Users in this group can perform basic network monitoring tasks.

SrAdmin
: Intended for system administrators or those with more network experience. Users in this group can perform all network monitoring tasks, including advanced network problem determination, configuration changes, and security administration.

When you create a NetView for AIX user profile, you specify the groups to which the user belongs. This process also enables you to create a new group, thus, defining a different set of access permissions. See “Managing NetView for AIX User Profiles” on page 27 for more detailed information.

As an example how group permissions work, consider the following groups and their associated security registration files:

Table 3. An Example of Group Permissions

User	Group	Security Registration Files
operator	Oper	ApplA ApplB
admin	SrAdmin	ApplA ApplB ApplC

The permissions defined for ApplA and ApplB are different depending on whether the user belongs to the Oper group or the SrAdmin group. The user, `admin`, in the SrAdmin group might have execute permission while the user, `operator`, in the Oper group might have read permission. A user in the Oper group does not have any access to ApplC; only a user in the SrAdmin group has permissions for ApplC.

Customized NetView for AIX Graphical Interface

When a user logs into NetView for AIX, the graphical interface presents only those functions and tools to which that user's group has access. For example, if a group has no permissions set for a specific menu operation, the menu operation is not displayed for that group of users. If a group has read permission set for a specific menu operation, the menu operation is grayed out for that group of users.

A user can belong to more than one group, allowing you to define access control based on multiple user roles and levels of experience.

Audit Management

Security services collects audit data for the following security-related activities:

- Configuration changes. These include:
  - Event configuration.
  - Changes to polling intervals.
  - SNMP configuration.
  - Configuration changes defined by vendor applications.

    Refer to the *NetView for AIX Programmer's Guide* for more information.
- Function access (element access)
- Login/logoff (includes Shift_in and Shift_out)

The audit data is stored in a log file, which you can review. This ability to review the audit data enables you to investigate any unusual activity that might indicate an attempted security breach. It also enables you to review normal usage patterns on your network and reliably trace security-related activities to a user, date, and time.

See “Reviewing Audit Data” on page 39 for information about how to view the audit data.

If you do not want to collect any audit data or if you do not want to collect data for all three categories, you can change this setting when you define the global security settings. See “Defining the Global Security Settings” on page 34 for more detailed information.

Consistent Security Controls

You can configure your security policy on one server and distribute the same security configuration to other servers in your network. Distribution of a central security configuration provides consistent security controls and reduces security-related administration tasks. You identify your manager station as the distribution server when you define the global security settings.

See “Distributing the Security Configuration” on page 38 and “Defining the Global Security Settings” on page 34 for more detailed information.

Pager Service For Event Correlation Rulesets

You can define pager information when you create a NetView for AIX user profile that is used to automatically issue a call to a pager. You do not need to turn security on to make use of the pager service.

When you create an event correlation ruleset that includes a paging action, you specify the NetView for AIX user ID of the person to be paged. The NetView for AIX program looks for the pager information in that user's profile. If the NetView for AIX user profile does not exist when you create the ruleset, a dialog box is displayed in which you can enter the user ID and pertinent pager information, and the user profile is created.

See “Creating and Editing a Ruleset” on page 128 and “Managing NetView for AIX User Profiles” on page 27 for more detailed information.

How To Define a Security Policy—An Overview

The following list describes the process you should follow to define your security policy:

1. Create a user profile for each NetView for AIX user. You might also want to create new NetView for AIX groups to create different sets of access permissions.
2. If you created new NetView for AIX groups, define the access permissions for the new groups.
3. Define the following global security settings.
4. Test your security configuration.
5. Turn security on.

Define your security policy using the Security Administration dialog box as shown in Figure 3 on page 25.

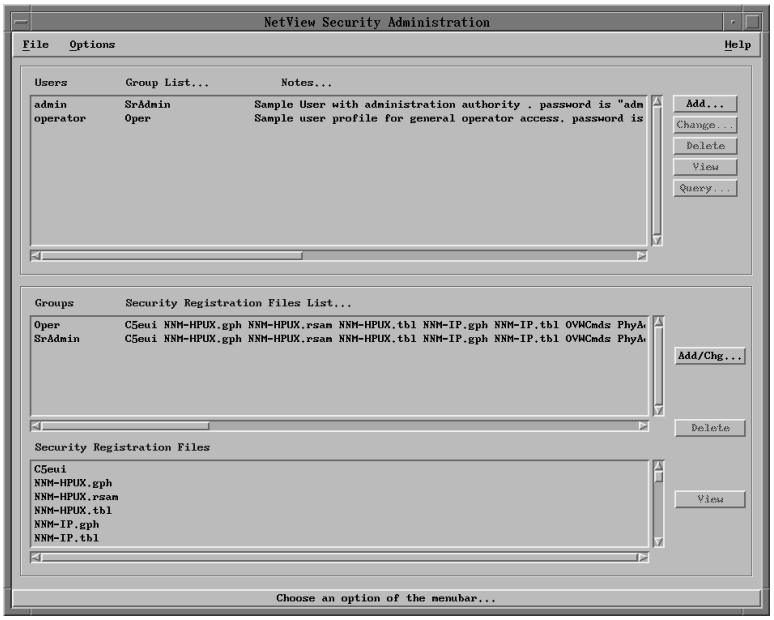


Figure 3. Security Administration Dialog Box

Accessing the Security Administration Dialog Box

By default, you must be a root user to access the Security Administration dialog box. After security is turned on, you must also be an authenticated NetView for AIX user; that is, you must log in with a valid NetView for AIX user and group ID and have NetView for AIX permission to execute the Administer..Security Administration operation (**nvsec_admin** command). The SrAdmin group is pre-configured with permission for security administration.

You can change AIX permissions so that non-root users can execute security administation. To do so, follow these steps as a root user on the NetView for AIX server that you want to use for security administration:

Step 1. Create a new AIX group or use an existing AIX group. The remaining steps use a group named `secadmin`.

Step 2. Set group permissions for the **nvsec_admin** command as follows:

```
chgrp secadmin /usr/OV/bin/nvsec_admin
chmod g+x /usr/OV/bin/nvsec_admin
```

Step 3. Set security AIX file permissions as follows:

```
chgrp -R secadmin /usr/OV/security

chmod g+rwx /usr/OV/security

chmod -R g+rw /usr/OV/security/C

chmod g+rx /usr/OV/security/conf

chmod g+rw /usr/OV/security/conf/sec.conf

chmod -R +rwx /usr/OV/security/C
```

Step 4. To execute security administation as a non-root user from a client machine, define the same AIX login ID on the server machine.

Step 5. Add the security administation group, for example, `secadmin`, to the groups set for root.

To access the Security Administration dialog, enter the **/usr/OV/bin/nvsec_admin** command in an aixterm window. When security is turned on, you can also access the Security Administration dialog box by selecting **Administer..Security Administration** from the NetView for AIX main menu.

If you want the **Administer..Security Administration** option to be selectable when security is turned off, edit the /usr/OV/registration/C/nvauth file and remove the `Security ;` line from the "nvsecadmin" Action clause.

Description of the Security Administration Dialog

The Security Administration dialog box contains three sections:

Users
: Lists all users known to NetView for AIX and the groups to which they belong. Two sample user IDs are provided: operator and admin. The passwords are the same as the user IDs: `operator` and `admin`, respectively.

  Note: You should change the passwords for the sample user IDs before you turn security on.

  You can use the buttons to add, change, delete, and view user profiles. You can also query logged on users.

Groups
: Lists all groups known to NetView for AIX and their associated security registration files. Initially, the Oper and SrAdmin groups are listed. You can use the buttons to add, change, copy, delete, and view group permissions.

Security Registration Files
: Lists all applications registered with security services. Select an application in the list and then select the **View** button to view the application's security registration file.

  Note: You can view only one security registration file at a time. If you have selected more than one security registration file, the **View** button is not available.

Refer to the *NetView for AIX Programmer's Guide* for information about security registration file syntax.

The menu options available when you select **Options** on the Security Administration menu bar enable you to define global security settings and perform other administrative tasks, such as, querying all logged in users, sending messages to logged in users, forcing off a logged in user, viewing audit reports, and distributing the security configuration to other servers in your network.

Managing NetView for AIX User Profiles

Security services authenticates each user who attempts to log into NetView for AIX and permits access based on the information in the user's profile. The user's profile includes login information, such as, the NetView for AIX user ID, the NetView for AIX groups to which the user belongs, and the user's password.

The groups to which the user belongs define the user's access permissions. When you create a NetView for AIX user profile, you can add the user to an existing group. To create a unique set of access permissions for a user, you can create a new group. You can update a user's profile any time, for example, to add the user to another group or include additional login controls, such as permissible login times.

Use the User dialog box as shown in Figure 4 on page 28 to perform the following types of tasks:

- "Creating and Changing a User Profile" on page 30
- "Viewing a User Profile" on page 30
- "Deleting a User Profile" on page 30
- "Adding and Changing a Group" on page 30
- "Copying a Group" on page 32
- "Deleting Group Permissions" on page 33
- "Viewing a Group's Permissions" on page 33
- "Deleting a Group" on page 33

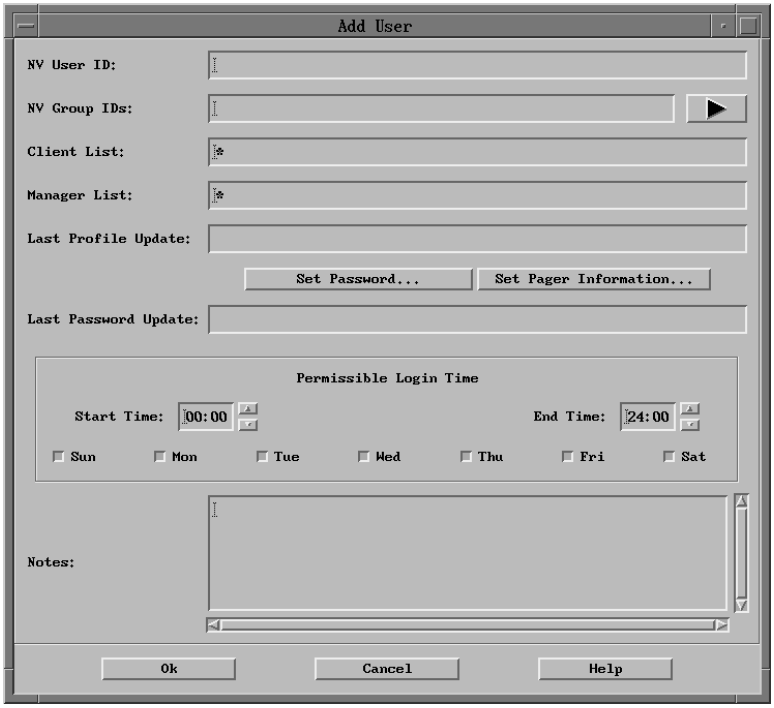


Figure 4. User Dialog Box

Description of the User Dialog Box

The User dialog box contains three sections. The top section contains general user profile information and contains the following fields and buttons:

NV User ID	Mandatory field that specifies the user's NetView for AIX login ID.
NV Group IDs	Mandatory field that specifies the group or groups to which the user belongs. Type or click on the arrow button next to this field to specify one or more group names. Separate multiple group names with a comma.
Client List	Optional field that specifies the client host names from which the user can log in. The asterisk (*) in this field indicates permission from all clients. Specify one or more host names to restrict login from the specified clients. Separate multiple host names with a comma.
Manager List	Optional field that specifies the management stations to which this user can log in. The asterisk (*) in this field indicates permission to log into any server. Specify one or more host names to restrict login to the specified servers. Separate multiple host names with a comma.

Last Profile Update	Informational field that indicates when the user profile was updated.
Set Password	Optional setting to set the user's password. You can change a user's password any time without knowing the user's current password. When you select the **Set Password** button, a password dialog is displayed. Enter the user's password in both password fields and select the **Ok** button. If the password matches in both password fields, the password is set and the dialog box closes. If you do not set the user's password, a default password is assigned. The default password is set to the same string as the user ID. For example, if you are creating a user profile for `pamb` and do not set a password, the password is set to `pamb`.
Set Pager Information	Optional setting to specify the user's pager information. You can create an event correlation ruleset that automatically issues a call to a user's pager. When you create a ruleset that includes a paging action, you specify the NetView for AIX user ID of the person to be paged. The NetView for AIX program looks for the pager information in the NetView for AIX user's profile. See “Creating and Editing a Ruleset” on page 128 for more information. When you select the **Set Pager Information** button, a dialog box is displayed. Enter the user's numeric or alphanumeric pager ID in the appropriate field and use the arrow button to select the carrier name. Select the **OK** button to close the dialog box.
Last Password Update	Informational field that indicates when the password was last updated.

User IDs, Group IDs, and passwords can be from 1 to 8 characters long and can consist of the letters a through z and A through Z in addition to the following characters:

`. , ; ( ) ' / - _ & + % = < >`

Note: Some of the allowed characters have special meaning to the AIX operating system. Avoid using those characters that have special meaning to AIX, especially at the beginning of an ID or password. For example, the < can be interpreted as input redirection and can cause errors when logging in from the command line.

Use the Permissible Login Time section if you want to restrict user login to a specific time and specific days. Type or use the arrow keys to set the appropriate start and end times and use the toggle buttons to select the appropriate days of the week. The default setting permits the user to log in any time on any day of the week.

Use the Notes section if you want to add comments to the user profile, such as the user's name and phone number.

Creating and Changing a User Profile

Use this procedure if the groups to which the user belongs are already created. If you want to create a new group for a user, use the procedure described in “Adding and Changing a Group” or “Copying a Group” on page 32.

Follow these steps:

Step 1. Access the Security Administration dialog box.

Step 2. Do one of the following:

- If you are adding a user profile, select the **Add** button next to the Users section.
- If you are changing a user profile, select the appropriate user in the Users section. Select the **Change** button.

The User dialog box is displayed as shown in Figure 4 on page 28.

Step 3. Make the appropriate changes to the dialog box. Use the online help if you need information about the dialog box fields. Select the **Ok** button.

The User dialog box closes, and the new user is added to the Users section of the Security Administration dialog box. The user has the permissions associated with the groups to which the user belongs.

Step 4. Repeat steps 2 through 3 for each user profile you want to add or change.

Viewing a User Profile

You can view user profile information by selecting the appropriate user ID in the Users section of the Security Administration dialog box. Then select the **View** button.

Deleting a User Profile

To delete a user profile, select the appropriate user from the Users section of the Security Administration dialog box and then select the **Delete** button. Select the **Delete** button on the Delete Confirmation box to confirm this action.

Adding and Changing a Group

Adding a group enables you to customize access permissions for a particular user or set of users. You can also add a group by copying an existing group's profile.

Minimally, you need to enable permission to run the graphical interface by setting read and execute permission for the ovw_binary and File SRFs. If you want to create a group that has minimal NetView for AIX security permissions (permission to run the graphical interface), you might find it useful to copy the `Oper` group and exclude permissions to SRFs that you do not want the group to access. See “Copying a Group” on page 32 for those steps.

Changing a group enables you to customize access permissions for all the users who belong to the group. When a new application registers with security services, for example, you would change the groups that should have access to the new application, defining the permissions each group should have. If you changed a group's permissions, users in that group will not get the change until they log off and then log back on again.

To create or change a group, follow these steps:

Step 1. Access the Security Administration dialog.

If you are changing a group, go to step 4. If you are creating a group, continue to the next step.

Step 2. Add a user to the group using one of the following methods:

- If the user profile is already created, select the appropriate user in the Users section of the Security Administration dialog box. Then select the **Change** button.

  or

- Create a user profile by selecting the **Add** button.

The User dialog box is displayed as shown in Figure 4 on page 28.

Step 3. Make the appropriate changes to the dialog box fields, adding the new group name to the NV Group IDs field. Then select the **Ok** button. Refer to the online help for detailed information about these fields.

The User section of the Security Administration dialog box is updated with the information, and the group name is added to the Groups section. At this point, the group does not have any access permissions.

Step 4. Select the group name in the Groups section of the Security Administration dialog box. Then select the **Add/Chg** button.

The Add/Change Group Security Registration dialog box is displayed.

Step 5. Click on the arrow button next to the Applications field and select every security registration file and select the **OK** button.

Note: By default, if you do not select a security registration file, users have access to the elements defined in that SRF and those menu options will be displayed.

Step 6. Select the **OK** button on the Add/Change Group Security Registration dialog box.

An Add/Change Group Security Registration dialog box is displayed that contains a list of each application's elements.

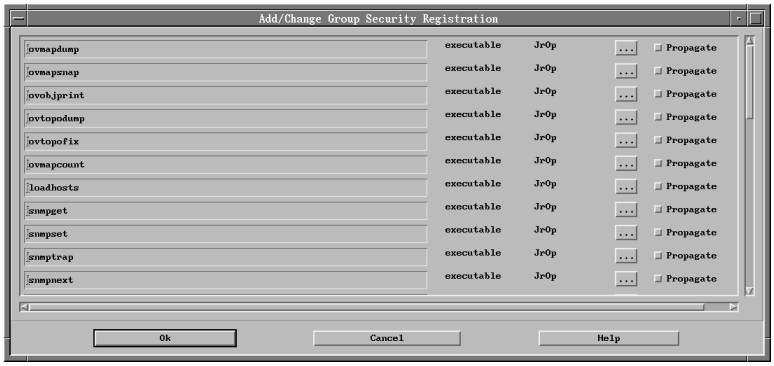


Figure 5. Add/Change Group Security Registration Dialog Box

Step 7. Click on the permissions box. 

The Element Permissions dialog box is displayed.

Step 8. Select the appropriate permission for the element and select the **OK** button:

- Select **r** to make the element unavailable. Menu options are displayed but are greyed out.
- Select **rx** to make the element available. Menu options are displayed and are available, and command line executables are available.
- Do not select any permissions to make the element unavailable. Menu options are not displayed.

Note: In general, write permission (**w**) is not used. Write permission is used only if an application's SRF file includes **w** as a valid permission.

Step 9. Select the **Propagate** button to propagate the permissions to all sub-elements (lower level menu options) or individually set the permissions for each sub-element as described in step 8.

Step 10. Select the **OK** button on the Add/Change Group Security Registration dialog box.

The group profile is updated with the permissions that you set.

Copying a Group

If you want to create a group based on an existing group's profile, you can copy a group. Then change the group to meet your requirements.

To copy a group, follow these steps:

Step 1. Access the Security Administration dialog box.

Step 2. Do one of the following:

- Add or change a user profile, entering the new NetView for AIX group name in the NetView for AIX Group IDs field. See “Creating and Changing a User Profile” on page 30 for those steps. Then select the

group name in the Groups section of the Security Administration dialog box and select the **Copy** button.

- Select the **Copy** button to copy a group without first adding a user to the group.

The Copy Group Profile and Permissions dialog box is displayed.

Step 3. Use the arrow button next to the Source Group field to select the group name that you want to copy.

Step 4. Enter the name of the new group in the Target field and select the **OK** button.

The group you have just created has the same permissions as the group you copied.

Step 5. Change the group to meet your requirements. See “Adding and Changing a Group” on page 30 for those steps.

Deleting Group Permissions

To delete group permissions, select the group name in the Groups section of the Security Administration dialog box, select the appropriate security registration file in the Security Registration Files Section, and select the **Delete** button. The security registration file you selected is deleted from the group's security registration files list.

Viewing a Group's Permissions

To view the permissions set for a group, select the group name in the Groups section of the Security Administration dialog box. Then select the appropriate security registration file in Security Registration Files section and select the **View** button.

Deleting a Group

Before you can delete a group, you must remove all users from the group. To do so, update the appropropriate users' profiles, deleting the group name from the NV Group IDs field. See “Creating and Changing a User Profile” on page 30 for those steps. When you delete the last user in the group, a delete confirmation dialog box is displayed. Select the **Delete** button to delete the group profile.

Setting User Environment Variables

You can set the environment variables, NVID and NVGID, in each user's .profile or .kshrc file so that the user's login ID and group ID are automatically displayed in the Authentication login panel.

For example, if Mary's login ID is mary in the SrAdmin group, add the following lines to Mary's .profile:

```
export NVID=mary

export NVGID=SrAdmin
```

Defining the Global Security Settings

To define the global security settings, use the Global Settings dialog box as shown in Figure 6.

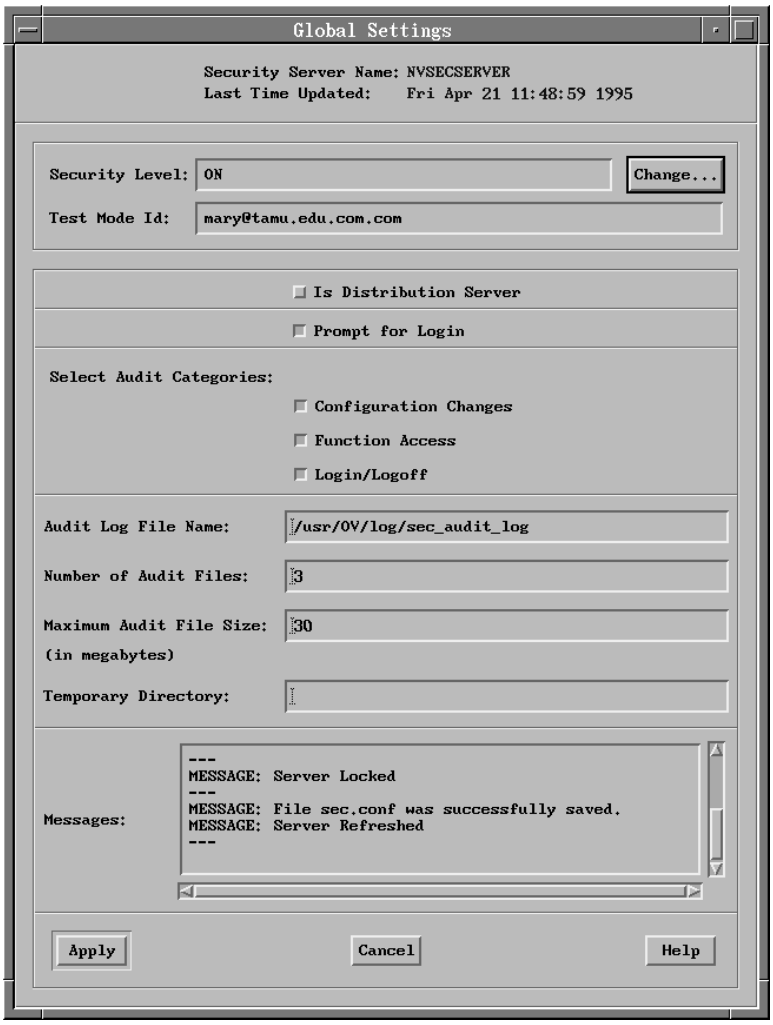


Figure 6. Global Settings Dialog Box

Description of the Global Settings Dialog Box

The Global Settings dialog box contains the following fields and buttons:

Security Level — Use the **Change** button to select one of the following settings:

ON — Turns security on for all users.

OFF
: Turns security off for all users and is the default setting.

TESTMODE
: Enables you to test your initial security configuration (before turning security on). Enter an AIX user ID and host name in the Test Mode ID field. You can log in as any NetView for AIX user in any NetView for AIX group from this AIX ID to test the configuration. Security is off for all users except for the test mode ID.

  Because TESTMODE has the same effect as turning security off, test future configuration changes by logging on with a NetView for AIX ID in the NetView for AIX group you want to test.

  Note: No other users can be logged into NetView for AIX when you change the security level. If other users are logged in, a message will be displayed. See "Managing Logged In Users" on page 37 for information about how to query who is logged on and communicate with logged-in users.

Is Distribution Server
: Specifies your NetView for AIX server as the security distribution server. When you select this button, you can distribute the security configuration to other servers in your network. See "Distributing the Security Configuration" on page 38 for more information.

Prompt for Login
: Specifies a password prompt when logging into NetView for AIX and is the default setting. If you have a third-party security product installed, you can turn this setting off so that your security product manages the login procedure, allowing users to log in using a single password.

Select Audit Categories
: Specifies the audit data you want to monitor. By default, audit data is collected for the following audit categories:

  - Configuration Changes
  - Function Access
  - Login/Logoff

  See "Audit Management" on page 23 for more information about collecting audit data.

  Data is logged in the log file you name in the Audit Log File Name field. You can review the audit data by using the

Options..Audit Report operation from the Security Administration dialog box menu bar. See “Reviewing Audit Data” on page 39 for more information.

Audit Log File Name
: Specifies the full path name of the audit log. The default audit log is the /usr/OV/log/sec_audit_log file.

Number of Audit Files
: Specifies the number of backup audit log files to keep; the default is **3**. When the audit log file reaches the size specified in the Maximum Audit File Size field, it is moved to a backup file in the directory that is specified in the Temporary Directory field. The naming convention for the backup audit log files is as follows:

  ```
  log_file_name.timestamp
  ```

  Where log_file_name is the name of the audit log file specified in the Audit Log File Name field, and timestamp indicates the date and time when the audit log file was backed up. The backup process continues until the specified number of audit files is reached. Then, the oldest backup audit log file is removed when the current audit log file is backed up.

Maximum Audit File Size
: Specifies the maximum audit log file size in megabytes; the default is **30 MB**.

Temporary Directory
: Specifies the full path name of the directory in which to store the backup audit log files.

Steps

To define the global security settings, follow these steps:

Step 1. Access the Security Administration dialog box.

Step 2. Select **Options..Global Settings** from the Security Administration menu bar.

The Global Settings dialog is displayed.

Step 3. Make the appropriate changes to the dialog box.

Step 4. Select **Apply** to apply the changes and close the Global Settings dialog box.

Managing Security

This section describes the administrative tasks you can perform that can help you enforce and manage your security policy. The following tasks are described:

- “Managing Logged In Users” on page 37
- “Distributing the Security Configuration” on page 38
- “Reviewing Audit Data” on page 39

Managing Logged In Users

You might find it necessary to send administrative messages, find out who is logged on, what processes are running, or force off a logged in user. For example, if a process is consuming system resources, you can find out who is running that process by querying logged in users. You can also send any administrative message to a logged in user.

Suppose you updated a group's permissions for a new application. Users in that group will not get the change until they log off and then log back on. You might take steps similar to the following:

Step 1. Access the Security Administration Dialog.

Step 2. Determine who is logged in using one of the following methods:

- Select **Options..Query All** on the Security Administration dialog menu bar.
- Select one or more users from the Users section on the Security Administration dialog. Then select the **Query** button.

The Logged In Users panel is displayed containing a list of logged in users.

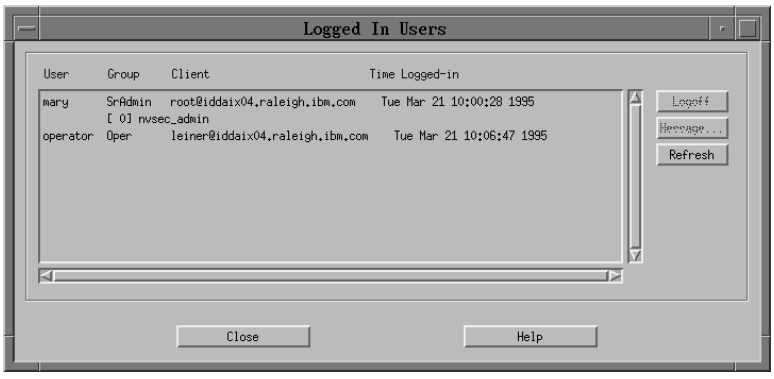


Figure 7. Logged In Users Dialog Box

Step 3. Send a message to the appropriate user informing the user to log off and then log back on to access the new application.

a. Select the appropriate user on the Logged In Users dialog box and then select the **Message** button.

A message dialog is displayed.

b. Type the appropriate message and select the **Send** button.

Step 4. If the user does not respond, you can log the user off by selecting the user ID on the Logged In Users panel. Then select the **Logoff** button.

The user is logged off, and a message is displayed informing you that this action is completed.

Distributing the Security Configuration

To distribute your security configuration, you must have defined your server as the distribution server on the Global Settings dialog box. See “Defining the Global Security Settings” on page 34 for those steps.

You can distribute your entire security configuration or selected file sets, depending on the configuration changes you have made. Follow these steps:

Step 1. Access the Security Administration Dialog box.

Step 2. Select **Options..Distribution** on the Security Administration dialog menu bar.

The Security Distribution dialog box is displayed.

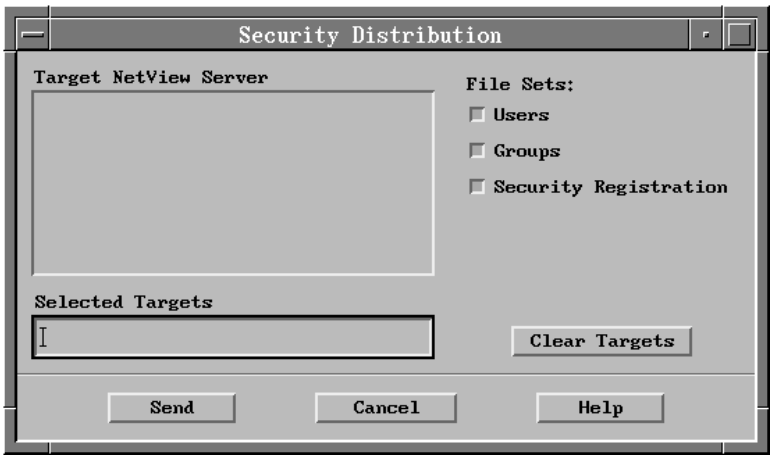


Figure 8. Security Distribution Dialog Box

Step 3. The section labeled Target NetView Server lists the managers that have been discovered. Select from the list or type the target server host names.

Step 4. Select one or more file sets.

The Users file set contains user profiles, the groups file set contains group profiles and group permissions, and the Security Registration file set contains the SRFs for each application registering with security services. After initially configuring your security policy, distribute all the file sets. Thereafter, when you make configuration changes, you can distribute only the file sets to which you made changes.

Step 5. Select **Send**.

The status window displays the transmission status for each target server, and the dialog box closes.

Security configuration files on the target servers are automatically backed up when you distribute new configuration files should you need to restore the prior configuration.

Each backup directory contains a time stamp (represented by *ddmmyytime*) indicating the day, month, year, and time when the directories were backed up. A list of the file sets and the backup directory for each on the target server is as follows:

File	Backup Directory
Users	/usr/OV/security/$LANG/Users.*ddmmyytime* If you need to restore this directory, copy the directory to the /usr/OV/security/$LANG/Users directory.
Groups	/usr/OV/security/$LANG/Groups.*ddmmyytime* If you need to restore this directory, copy the directory to the /usr/OV/security/$LANG/Groups directory.
Security Registration	Application SRFs: /usr/OV/security/$LANG/Domains.*ddmmyytime*/registration If you need to restore this directory, copy the directory to the /usr/OV/security/$LANG/Domains/registration directory. Group SRFs (group permissions) /usr/OV/security/$LANG/Domains.*ddmmyytime*/*groupname* If you need to restore this directory, copy the directory to the /usr/OV/security/$LANG/Domains/*groupname* directory, where *groupname* represents the name of the NetView for AIX group.

Reviewing Audit Data

To view or print the audit data you are collecting, follow these steps:

Step 1. Select **Options..Audit Report** from the Security Administration menu bar.

The Security Audit window is displayed.

Step 2. View the reports, using one of the following methods:

To view all audit reports:

- Select **View..All** from the Security Audit menu bar.

To view specific data:

a. Select **View..By Criteria** from the Security Audit menu bar. Then select **View..Set Criteria**.

The Set View Criteria dialog box is displayed.

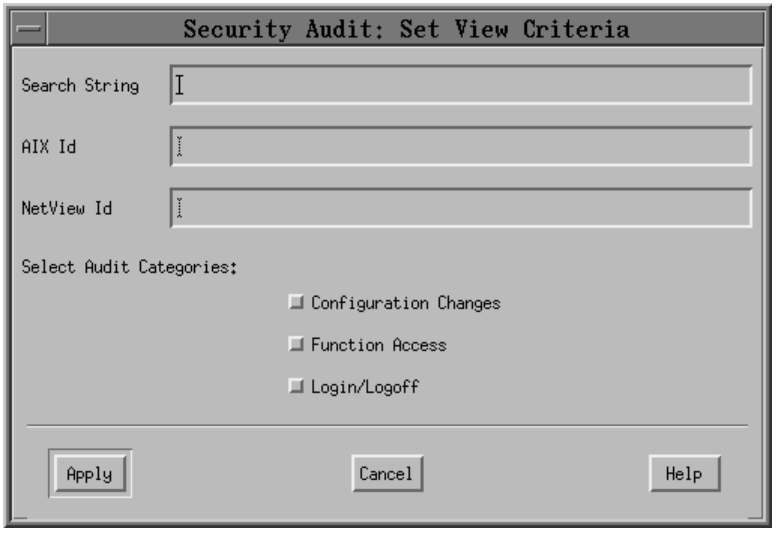


Figure 9. Set View Criteria Dialog Box

b. Enter the search criteria and select **Apply**.

The Open Audit File dialog box is displayed containing a list of directories that contain audit files.

When you select one or more directories, the list of files contained within those directories is displayed in the Files section.

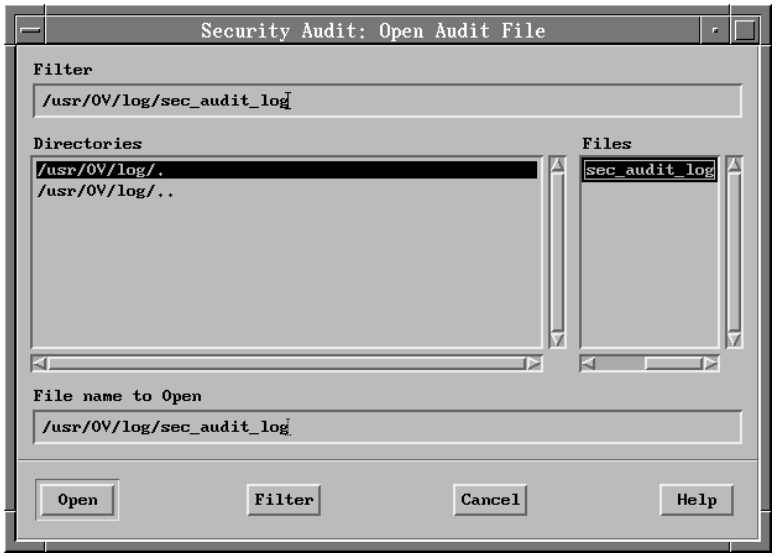


Figure 10. Open Audit File Dialog Box

Step 3. Select the files you want to view and select **Open**.

You can also open, print, or save a file using the operations available when you select **File** on the Security Audit window.

Converting ARFs to SRFs

To convert existing NetView for AIX application registration files (ARF) to NetView for AIX security registration file (SRF) syntax for registering sensitive security resources, use the **/usr/OV/bin/c_arf2srf** command. See the c_arf2srf man page for more information.

Refer to the *NetView for AIX Programmer's Guide* for information about SRF syntax and integrating applications with security services.

Verifying Security Permission For Shell Scripts

If you use shell scripts to run NetView for AIX executables, use the **/usr/OV/bin/vfy_access** command to verify security permission. If you have a shell script named `myscript` that invokes the **ovobjprint** command, add the following line to the script:

```
vfy_access ovobjprint
```

You can specify one or more NetView for AIX executables with the vfy_access command. See the vfy_access man page for more information.

Chapter 3. Creating and Customizing Submaps

This chapter describes the major components of the graphical interface and explains how these components work together to help you monitor and manage your network. This chapter also describes how to create a customized submap hierarchy and how to group together objects in your network. These tasks can help you monitor and manage your network more effectively.

The following topics are described in this chapter:

Objects—Basics

An object is an internal representation of a logical or physical entity or resource that exists somewhere in a computer network. An object is made up of a set of fields that specify all the characteristics of the object. Examples of resources represented by objects include:

- A computer node
- A software process on a computer
- An IP network

Most of the objects discovered and displayed by NetView for AIX are network objects. However, objects can also be created by users or by applications and integrated with the NetView for AIX program. For information about how users can add objects, see "Adding Symbols and Objects" on page 66. For information about how applications can add and manipulate objects, refer to the *NetView for AIX Programmer's Guide*.

Displaying the Object Database

Object and attribute information is stored in the object database. You can display the contents of the NetView for AIX object database with the **ovobjprint** command. For more information about this command, refer to the man page.

Symbols—Basics

A symbol is a graphical representation of an object as it is displayed on a submap of a particular map. Symbols are presentation elements; objects are underlying database elements that describe network entities like workstations, networks, and interface cards. Several symbols can represent the same object, even when the symbols are on different submaps.

Even though symbols represent objects, symbols can have some additional characteristics beyond those of the object they represent. These characteristics, or attributes, can vary among the different symbols representing a particular object.

Defining Symbol Characteristics

The following list describes characteristics of symbols in the NetView for AIX graphical interface:

Symbol type
: The symbol type consists of the symbol class, which specifies the outer shape of the symbol, and the symbol subclass, which specifies the graphic shown within the shape.

  The NetView for AIX program provides a variety of predefined symbols. To see the symbol types provided by the NetView for AIX program, select **Help..Legend** from the NetView for AIX main menu. If necessary, you can define new symbol types using Symbol Type Registration Files, which are described in the *NetView for AIX Programmer's Guide*.

Symbol variety
: A symbol can be an icon symbol, which is a two-dimensional picture, or a connection symbol, which is drawn as a line connecting two symbols. It is important to understand the difference between a connection symbol, which is a line that often represents an interface card object, and a connector symbol, which is a diamond shaped icon symbol that can represent a device like a bridge or router.

Symbol location
: A symbol can reside on either the application plane or the user plane of a submap. Submap planes are described in “Understanding Submap Planes” on page 59.

Symbol behavior
: Symbol behavior defines what happens when you double-click on the symbol.

  Explodable Symbols
  : When you double-click on an explodable symbol, a child submap is opened. This is the default behavior for symbols.

  Executable Symbols
  : When you double-click on an executable symbol, the program represented by that symbol is started.

    An executable symbol is displayed as a raised button on the submap. To see an example of an executable symbol, select **Help..Legend** from the NetView for AIX main menu.

Symbol label
: Each symbol has a label that describes the object represented by the symbol. The label is displayed below the symbol. Because the NetView for AIX program does not use the symbol label to identify the symbol, the symbol label does not have to be unique. You can choose whether or not to display

the label of a symbol. See “Modifying and Displaying Symbol Labels” on page 95 for steps on displaying symbol labels.

Symbol status
: Symbols can display information about the status of the object or connection that the symbol represents. Colors are used to represent status information, as described in *NetView for AIX User's Guide for Beginners*.

Indicating Symbol Status

Symbol status conveys the current state of a network entity based on a predetermined set of rules. The graphical interface indicates the status of a symbol by its color on a submap. The status displayed by the symbol stems from the state of certain attributes of the object that the symbol represents. If the symbol represents a container object, for example an Internet symbol, its status color represents the combined status of all the symbols contained within it. See “Understanding Compound Status Source” on page 46 for information about compound status.

The ipmap application determines the status of a node object based on the operational status of all of the IP interfaces installed in the node. If all of a node's IP interfaces are down, the ipmap application reflects the node's status as critical. From the perspective of the ipmap application, the node is nonfunctional. However, another application might consider the same node to be running, because that application monitors a different protocol with fully functional interfaces. This is an example where you might have different symbols representing different states of one object.

The following list summarizes the status source used to determine status for the various symbols:

Object Status
: Interface card symbols and connection symbols that correspond to interfaces (not networks) derive their status by object status source.

Symbol Status
: All node symbols (connectors, servers, and computers) on network and segment submaps derive their status by symbol status source.

Compound Status
: All other symbols on your IP submaps derive their status by compound status source. These symbols include, but are not limited to:

  - All network and segment symbols
  - All location and internet symbols
  - All node symbols on the internet and location submaps

You can determine or change the symbol's status source by selecting **Edit Modify/Describe Symbol** from the object context menu.

Each status source is described in the following sections.

Note: If a symbol has object status source or symbol status source and users have configured events as status events, the Event Display application can change

the symbol's status. In addition, other applications can change a symbol's status. Refer to the *NetView for AIX Programmer's Guide* for more information.

By default, if a symbol has compound status source, the symbol's status is updated by the ipmap application. To enable the Event Display application to update a symbol's status if the status source is compound, change the value for the overrideCompoundStatus resource in the /usr/OV/app-defaults/Nvevents file to TRUE. When the value for the overrideCompoundStatus resource is TRUE, the Event Display application updates all symbols for the object without having to manually change the status source for each symbol.

Understanding Object Status Source

Object status source determines the symbol's status based solely on the status of an individual object in the object database. If the interface card object is the object database is down, then the status is reflected as critical. If the interface card object is up, then the status is reflected as normal. The status of the interface card symbol is based on that card object only.

Understanding Symbol Status Source

Symbol status source determines the symbol's status based on some algorithm different from object status source and compound status source. The algorithm is contained within the application that manages the symbol. For example, if an IP symbol has symbol status, then the ipmap application determines with its own algorithms what the status of the symbol should be. By default, the ipmap application uses symbol status for the following symbols:

- Any node symbol (connectors, servers, and computers) on a network submap

  The status of the connector is based on the combined status of only the interface cards that the connector has in that network, rather than the combined status of all of the interfaces in the connector.

- Any node symbols (connectors, servers, and computers) on segment submaps

  The status of the node is based on the combined status of only the interface that the node has in that segment, rather than the combined status of all of the interfaces in the node.

Some HP hubs, however, have their own status propagation rules because these HP hubs contain non-IP interface cards that do not report their status. The status propagation rules for these HP hubs are:

- On network submaps, the status of the hub symbol is based on the combined status of only the IP interface cards in that network.
- On segment submaps, the hub symbol's status is determined by compound status rather than symbol status.

Understanding Compound Status Source

Compound status source determines how status values are propagated from symbols in child submaps to the symbol on the parent submap that represents the child submap

and is based on the combined status of all of the symbols in that symbol's child submap. For example, the status of a segment symbol is based on the combined status of all of the node symbols in that segment's child submap.

The NetView for AIX program uses the following rules to determine the combined status of a group of symbols:

- Default compound status
- Propagate most critical compound status
- Propagate at threshold value compound status

Default Compound Status: The default compound status scheme determines compound status as follows:

Table 4. NetView for AIX Default Compound Status Scheme

A symbol's status is:	**If the symbols in the child submap meet these criteria:**
Normal	All symbols are either normal, acknowledged, UserStatus1, unmanaged, or unknown and At least one symbol is normal, acknowledged, or UserStatus1
Critical	At least one symbol is critical or UserStatus2 and no symbols are normal, acknowledged, or UserStatus1.
Marginal	Any of the following: • At least one symbol is normal, acknowledged, or UserStatus1 and At least one symbol is marginal, critical, or UserStatus2. • All symbols are marginal, unmanaged, or unknown and At least one symbol is marginal.
Unknown	All symbols are unknown or unmanaged.
Acknowledged	All symbols are acknowledged.

A symbol with compound status source will never have its status changed to UserStatus1, UserStatus2, or Unmanaged as a result of compound status propagation.

Propagate Most Critical Compound Status: The propagate most critical status scheme causes the graphical interface to propagate the status of the most critical symbol in the child submap to the submap's symbol in the parent submap.

Propagate at Threshold Value Compound Status: The propagate at threshold value (0%–100%) scheme enables you to set two threshold values, marginal and critical. The marginal threshold value determines when a symbol's status changes from normal to marginal. The critical threshold value determines when a symbol's status changes from marginal to critical.

- Conditions for marginal status:

  Marginal Threshold Value < % (Marginal + Critical) < Critical Threshold Value

If the percentage of symbols, that are either marginal or critical, in a parent object's child submap is greater than the marginal threshold value and less than the critical threshold value, then the parent object's symbols that have compound status source will be marginal.

If the percentage of symbols, that are either marginal or critical, in a parent object's child submap is less than the marginal threshold value, then the parent object's symbols that have compound status source will be normal.

- Conditions for Critical Status:

  Critical Threshold Value < % (Marginal + Critical)

  If the percentage of symbols, that are either marginal or critical, in a parent object's child submap is greater than the critical threshold, then the parent object's symbols that have compound status source will be critical. If the critical threshold value is less than the marginal threshold value, then symbols that have compound status source will change directly from the normal state to the critical state.

Setting Compound Status: If you have a map open with read-write authorization, you can change the default compound status scheme on the current map or when you create a new map. The status setting applies to the entire map. You can't set compound status scheme for individual submaps.

When you create a new map, you can set the compound status by selecting one of the following Compound Status buttons on the New Map dialog box:

- Default
- Propagate most critical
- Propagate at threshold values

Once the map is created, you can change the status scheme by selecting **Modify/Describe..Map** from the Edit pull-down menu. Select the preferred compound status button and select **OK**.

Maps—Basics

A map is a collection of objects stored in a map database that describe a set of network entities. They are displayed by the graphical interface for NetView for AIX. The graphical interface uses these objects to draw icon symbols and connection symbols on the map's submaps. A submap is a view, or window, that displays information stored in the map database. You do not view a map directly; instead, you view submaps that contain symbols representing objects within the map. Submaps are described in “Submaps—Basics” on page 54.

You can create, delete, or choose a map to be displayed from existing maps. You can even create several maps and control which applications operate on these maps. While you create maps and define their scope, applications dynamically update maps to reflect the state of the management environment.

Although you can create several maps, the graphical interface can have only one map open at a time. The open map is the map currently displayed by the graphical interface. This map can be updated by applications and users.

To view multiple maps simultaneously, you must invoke multiple instances of the graphical interface (ovw application). If there are multiple simultaneous ovw sessions with the same map open, only one session can have read-write access to the map.

Information in the Map Database

If you have a read-write map, select **Edit..Modify/Describe..Map** to change information about the map. Each map has the following attributes:

Name
: The name of the map, which is assigned when the map is created. Each name must be unique. You can change the map name.

Root submap
: The highest-level submap of the map. The root submap cannot be deleted.

Home submap
: The submap that is displayed in the initial submap window when you open the map. You can assign any submap of the map as the home submap. The root submap is the default home submap.

Layout algorithm for root submap
: The layout algorithm used for the root submap. The default is row/column. Once set, the layout algorithm cannot be changed.

Compound status scheme
: The compound status scheme that applies to the entire map. This scheme determines how the graphical interface propagates status from symbols in child submaps to the symbol of the parent object. You can change the compound status scheme for a map. You cannot change it for only a submap. See “Understanding Compound Status Source” on page 46.

Configurable applications
: Any configurable map applications that are available on your system for that map. You can enable or disable these when you create a new map.

Comments
: Any comments or notes about the map. This entry can be used to document the map's creation date, purpose, or other information you want to store.

Managing Maps in a Distributed Network Environment

In a distributed network environment (client/server), the NetView for AIX daemons run on the server, and the graphical interface applications run on the client. The client obtains event and topology status information from the daemons running on the server. A client/server configuration enables you to distribute the CPU and memory requirements to the client. You can use several clients, enabling you to divide the management tasks among more operators at one time.

The NetView for AIX graphical interface application maintains the map database. In other words, a client application maintains the map database. Whether the map database resides on the server or the client depends on how the client was configured. The client can be configured to NFS mount the map database from the server, or it can be configured to store the map database locally. Refer to *NetView for AIX Installation and Configuration* for more information.

Each type of configuration has certain advantages, and each presents different network management considerations.

When the Map Database is NFS Mounted

When the map database is NFS mounted, maps created on the client are stored through NFS on the server, and all users view the same set of map information. This is consistent with previous versions of NetView for AIX regarding map information. The tradeoff is network performance. Retrieving map database information from the server depends on network speed, bandwidth, and the size of the discovered network. Some map operations will be slower than if the maps were stored locally.

Improving Network Performance for an NFS-Configured Client: To decrease the impact on network performance due to retrieving map database information, use the following methods:

- Create new maps on the server rather than on the client. When the map is fully created, it can be utilized from the client. If you create the map on the client, the client contacts the daemons on the server to determine the network topology and then, through NFS, writes the information back to the server to create the map, symbols, and submaps. Creating maps on the server minimizes the amount of network traffic required to create the map.
- Investigate using read-only maps on the client. NFS write operations will not be performed, and NFS caches read operations, which improves network performance.

Updating the Host Name For NFS Mounted Maps: Because the map names and their host name locations are stored in the object database, if you change the host name of the server, you must update the host name in the object database so you can access the maps.

To update the host name of the server when all maps are on the same machine (NFS mounted), use the following command:

```
mapadmin -u newhostname
```

Replace `newhostname` with the new name of the server.

See the mapadmin man page for more information.

When the Map Database is Stored Locally

When the map database is stored locally (on the client), better performance is obtained in retrieving map database information. However, because each client can have its own set of maps, more map administration is required. Consider the following:

- Changes you make to maps on the server are not reflected in maps on the client. If you want to make a change to all the maps, such as deleting a router, you must delete the router from each of the client maps.
- Restarting topology discovery on the server does not affect maps that may exist on the client. If the map on the client is out of date because the server's discovery has been restarted, the map is invalid and cannot be opened. You should delete invalid maps on the client. See “Deleting an Invalid Map on the Client” for more information.

Map names must be unique across the server's domain, because objects get stored in the object database based on a map's name. If a user tries to create a map with a map name that already exists on another client or on the server, the user is prompted to choose a different name.

In addition, clients and servers need to be configured with similar host name resolution procedures. Clients need to be able to resolve host name servers and other clients. If a domain name server (DNS) is used for a client or server, then DNS should be used for host name configuration for all clients and servers. Otherwise, a client can create a map and have its DNS-configured name recorded as the map owner, but a non-DNS client cannot resolve the host name.

Deleting an Invalid Map on the Client: If a map that is stored locally on the client is out of date with the map on the server, the map is invalid and cannot be opened. To delete the invalid map, use the following command:

```
mapadmin -r mapname
```

Replace `mapname` with the name of map that you want to delete.

You can use the **mapadmin -l** command to list all maps on the client. See the mapadmin man page for more information.

Updating the Host Name For Local Maps: Because the map names and their host name locations are stored in the object database, if you change the host name of the server or the client, you must update the host name in the object database so you can access the maps.

To update the host name when maps are stored on multiple machines (clients), you can use the **ovwls** to list all maps that exist and where they are located. Then use the **mapadmin -u mapname:newhostname** command to update the host name.

Assume that the **ovwls** command shows the following maps and locations:

```
Map Name       Location

default        nm014.raleigh.ibm.com
default_cs     cs.raleigh.ibm.com
mymap          cs.raleigh.ibm.com
```

If you changed the host name for `cs.raleigh.ibm.com` to `mgr1.raleigh.ibm.com`, use the following commands to update the host names for the maps named `default_cs` and `mymap`:

```
mapadmin -u default_cs:mgr1.raleigh.ibm.com

mapadmin -u mymap:mgr1.raleigh.ibm.com
```

Removing Maps on the Client: Before removing a server connection from a client or deinstalling NetView for AIX from the client, use the File..Delete Map operation or the **ovw -rmmap** *mapname* **command** to remove all maps that are stored on the client. If you do not remove the maps, information is left in the object database on the server regarding the number of maps that exist and where those maps reside. This can provide an incorrect picture of the maps that exist. Refer to *NetView for AIX Installation and Configuration* for information about removing client access and deinstalling clients.

Customizing Maps

You can customize maps to meet the needs of individual users. Customizing a map enables you to:

- Allocate responsibility for managing your network among several people. For example, network administrators expert in managing routers and gateways can open a map that is configured to help manage those devices.
- Use network management applications that perform a specific type of task, for example, data traffic or performance monitoring.
- Create a map that is a projection of an administrator's responsibility or sphere of influence.

You can customize the display of object information for maps you create. Several maps can display information about the same object because maps get their information from the same source, the ovwdb object database.

Reasons for Creating Several Network Maps

You can choose to create several maps of your network. The following are reasons for creating more than one map of a network.

- Sphere of influence

  You might want to have different maps for different areas of responsibility within your network. In large or complex networks, it can become impractical for one map to display all information about systems in your network. You can create maps that focus on a specific set of system or nodal capabilities. For example, one map might be concerned with software maintenance, and another map with bridge management.

- Management region

  You might want to manage a specific portion of your network or partition information about your network. Maps can have specific constraints and characteristics, run different applications, cover different geographic areas, and set compound status differently.

- Security

  You might want to provide various levels of access to information about your network for security purposes. Using NetView for AIX permissions, you can create a map of your network that cannot be edited, and another map that is identical to the first map except that it can be edited.

- Troubleshooting

  You might want to save a map so you can restore the map later, if necessary. It helps to keep a copy of a map before making changes that may have uncertain results.

- Customized maps

  You might want to combine the sphere of influence and management region to create a customized map of your network. Users can share the same map, or you can create a map for each user that combines specific management regions and specific spheres of influence. Perhaps a user wants several maps of the same network, each focusing on a different domain or organization. You can customize a map so that it is a projection of your responsibility for systems on your network. You can customize each map to display certain aspects of your network and avoid sifting through data you do not need. You can also use different background graphics.

- System-specific maps

  You might want to have separate maps for different kinds of systems on your network. For example, you might want one map for your IP systems, one map for OSI-based systems, one map for Apollo Domain systems, one map for NFS**, and one map for diskless systems.

Assigning Map Access Levels

Assigning map access levels enables you to limit or deny access on a per-user, per-map basis. Each user of your system will have one of the following types of access to each map:

No access	The user cannot open this map.
Read-only access	The user can see status changes, perform locate operations on objects, and update topological changes using the File..Refresh Map operation. However, the user cannot add, delete, or modify symbols, objects, or submaps.
Read-write access	The user can add objects, add connections, create submaps, and change object attribute values.

Changing Map Permissions

Only one user can have a specific map open with read-write access at one time. If you have a map open with read-write access and another user displays the same map, the other user's map is open with read-only access. This is the case even if the other user has read-write permission to the map. Several users cannot have simultaneous read-write access to a specific map. However, if each user creates a copy of this map, they can have read-write access to the copy.

There are several commands that you can use to assign users read-write or read-only access to a map. To change map permissions, use the **ovwperms** command or one of its convenience routines:

ovwls	Lists current permissions for specified maps.
ovwchown	Changes the owner of one or more maps.
ovwchgrp	Changes the group ID of one or more maps.
ovwchmod	Changes map permissions by mode. Do not use the file permission commands from your operating system.

For more information about these commands, refer to the *NetView for AIX Administrator's Reference* or the man pages. For steps on changing the map permissions using SMIT, see “Setting Map Permissions” on page 96.

Map Snapshots

A snapshot is a static image of a particular map that preserves the status of all symbols and contains all submaps that existed in the map at the time the snapshot was taken. Although you must take snapshots from a read-write map, the snapshots themselves are read-only and cannot be updated by applications. Select **File..Map Snapshot..Create** to take a snapshot of a map.

Use snapshots to document your network or to keep a record of your network's current status. It is useful to take snapshots before making major configuration changes. Select **File..Map Snapshot..Open** to open a previously created snapshot of a map.

When you open a snapshot, the home submap at the time the snapshot was taken is displayed. Only one snapshot can be open at a time; however, you can display the open map and a snapshot in submap windows at the same time. The name of the snapshot is displayed in the status line.

Although you can display an open map and a snapshot of the map at the same time, you can't perform the same operations on them. Operations that highlight a symbol on the open map do not highlight the same symbol on the snapshot. Highlighting objects applies only to the map or snapshot in which you highlight them.

Submaps—Basics

A submap is a collection of related symbols that are displayed in a single window. Each submap displays a different perspective of the information in the map. Typically,

submaps are organized in a hierarchy that enables you to see your network from a distance or to choose a more detailed view. You can customize the organization of submaps in a map to suit your purposes.

The most common method used to navigate through submaps is double-clicking the mouse on explodable symbols. Double-clicking on an explodable symbol causes a submap to be displayed if a submap is associated with the object the symbol represents. The object associated with the explodable symbol is called the parent object. The submap that is displayed by double-clicking on the symbol associated with the parent object is called a child submap. You can display more than one submap window at one time. To do so, holding the second mouse button, select a symbol and drag the symbol to an area outside the current submap. You can also use the Locate..Submap menu option to view a submap.

Working with Submaps

Submaps enable you to:

- Create a selective view into part of the management domain of a network.
- Choose a collection of symbols to display in a single submap window.

You can create, delete, and modify the characteristics of submaps in the open map. You might want to create a submap to display a detailed view of systems on your network. You can create increasingly detailed submaps of your network map. If you have a submap that becomes too congested, you can create a new submap and partition the information.

Using the Root Submap

The graphical interface creates a root submap that provides a standard, top-level submap on which you can display the symbols that represent different protocol views of the map. For example, on the root submap you will probably have an IP Internet symbol that represents all of the IP entities in your management domain. You could also have symbols that represent other types of topology entities. The root submap enables you to place more than one protocol symbol within one map.

Network and systems management applications can use the root submap to build hierarchies of submaps. The root submap serves as an anchor on which applications can place symbols that represent protocols. You can select one of these symbols and display the highest level of a submap hierarchy.

Using the Home Submap

Each map has a submap designated as the home submap. The home submap is the first submap that is displayed when the map is opened. You can assign any submap in the map as the home submap for all users of that map. By default, when a map is created, the root submap is designated as the home submap. If you delete the home submap, the root submap becomes the home submap until another home submap is assigned. See “Assigning a Home Submap” on page 80 for more information.

Submap Characteristics

Table 5 describes submap characteristics.

Table 5. Submap Characteristics

Characteristic	Description
Name	The name of the submap, as assigned when the submap was created. Each name must be unique within the scope of the map.
Parent Object	The object considered a parent object of a submap. The symbols representing the object explode into the submap. Because several symbols from different submaps can represent the same object, you can navigate to a child submap from several submaps. A submap might not have a parent object.
Parent Submap	The submap chosen as the parent of the current submap. During submap creation, a user or application can choose a submap to be the parent submap.
Layout	The layout algorithm used for the submap. Once the algorithm is set, it cannot be changed for an existing submap.
Presentation	The presentation format used by the NetView for AIX program to display the submap. The presentation can be either scaling or zooming.
Background Graphics	The graphic displayed on the background of the submap plane to customize the appearance of the submap.
Comments	The comments, notes, or keywords about the submap.

Creating Child Submaps and Independent Submaps

In the NetView for AIX program, there are child submaps and independent submaps. The method you use to create a new submap determines whether the submap is a child submap or an independent submap (also known as an orphan submap).

A child submap represents a detailed view of its parent object. To create a child submap, double-click on an explodable symbol whose parent object has no child submap. For more information about creating child submaps, see “Creating a Child Submap” on page 76.

An independent submap has no parent object or parent submap. To create an independent submap, select **Edit..Create Submap**. To display an independent submap, select the submap from the Submaps in Map dialog box and select **Open Submap**, or select the independent submap from the Navigation Tree window.

Understanding Submap Presentation

Submap presentation enables you to choose how each submap presents symbols and background graphics. You can choose between scaling and zooming for each specific submap.

Scaling

Scaling enables you to display an overall view of the submap. All displayed symbols and the background graphic scale to the size of the submap window. Scaling is the default setting when a submap is created.

Zooming

Zooming enables you to display a close-up view of the submap. Scroll bars are displayed when only a portion of the submap fits into the viewing area. If you are working with a read-write submap and you use the scroll bars to change the visible portion of the submap, the layout is saved.

You can use the zoom feature using the following methods:

- Set a zoom factor to determine the extent to which you zoom into the view of the submap. The default zoom factor is **one**.
- Use quick zoom to draw a boundary box around the area of the submap to be magnified.

To set a zoom factor:

1. Select **Modify/Describe** from the Edit pull-down menu.
2. Select **Submap**.
3. Select the Zooming button and use the slider bar to select a zoom factor.
4. Select **OK**.

To use quick zoom:

1. Position the mouse cursor in the upper left corner of the area to be magnified.
2. Press and hold down the Shift key and mouse button 1. Then drag the mouse to draw a box around the area to be magnified.
3. Release mouse button 1, then the Shift key. If you release the Shift key first, the objects will be selected instead of magnified. Use the Locate..Selected Objects List..Deselect All menu operation to deselect them and try again.

To return to the original presentation, press and hold down the Shift key and click mouse button 1 anywhere on the map.

Displaying a submap with background graphics and zoom presentation uses a large amount of memory. This is because the entire virtual display has to be buffered, enabling the user to navigate using the slide bars. The checkZoomAllocation resource in the /usr/OV/app-defaults/OVw file controls whether a warning dialog box is displayed every time a user zooms in on a submap with background graphics. Zoom ratios greater than the value set for checkZoomAllocation cause a warning dialog box to be displayed. Set the resource to 10 if you do not want a warning dialog box to be displayed. Set the resource to 0 if you want a warning dialog box to be displayed for any zoom ratio.

See “Changing the Graphical Interface Defaults” on page 98 for more information.

Understanding Submap Layouts

The way the NetView for AIX program arranges symbols on a submap is called the submap layout. The method for arranging symbols on the submap is called the layout algorithm. Symbols can be automatically placed on a submap as determined by the layout algorithm or they can be manually placed by the user.

Using Layout Algorithms

Each submap has an assigned layout algorithm that determines how symbols are arranged on the submap. The layout algorithms are based on common network topologies. Table 6 lists the available layout algorithms.

Table 6. Network Topology Layout Algorithms

Algorithm	Arrangement on the Submap
Row/Column	Symbols are arranged in rows and columns.
Point-To-Point	Symbols are arranged as an arbitrarily inter-connected set of nodes and connections.
Bus	Symbols are arranged along a backbone representing the linear array of nodes on a segment.
Star	Symbols are arranged in a star consisting of a circle and a center symbol. You can set the star center using the symbol pop-up menu.
Ring	Symbols are arranged in a circle.
Tree	Symbols are arranged in a hierarchical tree structure.
No Layout	Symbols are arranged by the user or are left in the New Object Holding Area.

You can set the layout algorithm for a submap only when the submap is created. The layout algorithm cannot be changed for that submap after it has been created. If an application creates a submap, it can specify a layout algorithm. If no layout algorithm is specified when the map is created, a default layout algorithm is selected. The default layout algorithm for a submap is based on the symbol type of the parent object.

Note: You cannot change the layout algorithm for any submap of the system default map. The default layout algorithm for the root submap is row/column.

Using Automatic Layout

Automatic layout either enables or disables enforcement of the layout algorithm of a submap. You can enable or disable the automatic layout algorithm for a selected submap. After enabling or disabling automatic layout for all submaps in the open map, you can change the automatic layout setting to on or off for all submaps. See “Setting Automatic Layout” on page 95 for steps on setting automatic layout.

Using the New Object Holding Area

A New Object Holding Area is displayed in the lower portion of each submap window if the submap has no layout algorithm or if automatic layout is disabled for that submap and new objects have been discovered. Symbols in the New Object Holding Area are

shown without their connections. To drag symbols from the new object holding area and place them in the associated submap, hold the Ctrl key and using mouse button 2, select the symbol, drag it to the submap, and release the mouse button.

Understanding Submap Planes

Submaps contain the following three layers, or *planes*:

- Background plane
- Application plane
- User plane

The *background plane* provides the background against which symbols are viewed. You can add background graphics in the background plane to provide a context for looking at symbols in your submap presentation. Symbols on the application plane and the user plane appear on top of the graphic. For more information about adding background graphics, see “Adding or Removing a Background Graphic” on page 93.

Symbols on the *application plane* represent objects that are managed by at least one network or system management application. If one or more applications manage an object, one or more symbols of that object are displayed on the application plane. The symbols of that object are presented flat against the submap background. If no applications are managing the object, all symbols representing that object appear on the user plane.

Symbols on the *user plane* represent objects that are created by users and not managed by any applications. The NetView for AIX program distinguishes symbols on the user plane by providing a shadow for symbols to make them appear raised above the submap background.

The Metaconnection Submap

A metaconnection symbol represents more than one connection between two symbols or a symbol and a backbone on a submap. For example, suppose a gateway has more than one interface card in the same network. On the IP Internet submap, the connection between the gateway and the network is a metaconnection. This is because the connection symbol between the gateway and the network represents two connections (the two interface cards). When you double-click on a metaconnection symbol, the metaconnection submap opens.

The metaconnection submap displays the status of each connection between the two symbols. The graphical interface creates a metaconnection submap when a user or an application adds a second connection between two symbols or a symbol and a backbone.

You can add an unlimited number of connections between two symbols in a regular submap. Each of these connections is automatically added to the metaconnection submap with the two symbols or the symbol and the backbone as end points.

It is possible to create a regular child submap for a connection object, if only one connection exists. If that connection becomes multiple, a metaconnection submap is

created. You can no longer access the regular child submap by double-clicking on the connection. However, you can select the regular child submap from the Submaps in the Map dialog box. You cannot see individual connections between the two symbols on the submap in which the metaconnection symbol is displayed.

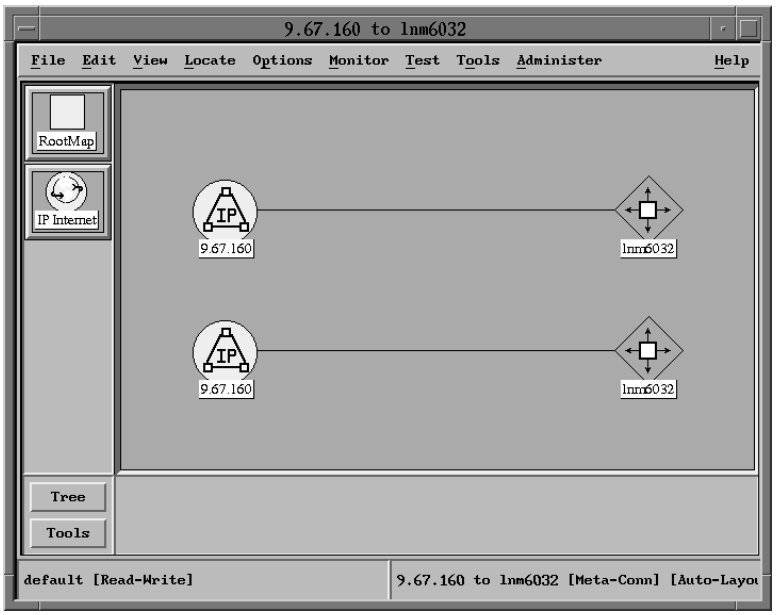


Figure 11. Example of a Metaconnection Submap

Characteristics of a Metaconnection Submap

A metaconnection submap has the following characteristics:

- Displays all the connections represented by the metaconnection symbol.
- Has a row/column layout unless the connections are between a symbol and a backbone.
- Displays the two end points of the connection in the metaconnection submap for each connection in the submap.

Behavior of a Metaconnection Submap

The behavior of the metaconnection submap is similar to a regular submap in some ways:

- You can create a child submap from a metaconnection submap by double-clicking on any of the objects in the metaconnection submap. Therefore, the metaconnection submap can be a parent of other regular submaps.
- You can select objects in the metaconnection submap.
- You can add unconnected objects to a metaconnection submap.

The behavior of the metaconnection submap differs from a regular submap in some ways:

- You cannot add connections to a metaconnection submap.
- You cannot delete the last object from the submap (whether it represents a connection or not) without deleting the metaconnection and metaconnection submap.
- You cannot see propagated status for connected symbols. Compound Status works differently in metaconnection submaps. The metaconnection symbol displays the compound status of the multiple connections in the metaconnection submap. Any unconnected objects in the metaconnection submap also contribute to compound status. However, the connected icon symbol in a metaconnection submap do not propagate their status. Their status is maintained by the symbols in the parent submap above the metaconnection submap.

Using NetView for AIX Applications

An *application* is a program that interacts with users through the graphical interface. Applications enable you to perform the following actions:

- Process user requests
- Create or delete objects, symbols, and submaps
- Change the contents of maps
- Provide special display functions

You can write and integrate applications with the NetView for AIX program. Refer to the *NetView for AIX Programmer's Guide* for more information about writing applications.

The ipmap Application

The ipmap application is the primary application used by the NetView for AIX program. When the graphical interface is started, the ipmap application is automatically started. The ipmap application ensures that the ovw application (the graphical interface) and the ovtopmd daemon (the IP topology database daemon) behave consistently.

For example, when an object is deleted using the graphical interface, the ovw application tells ipmap which symbols and objects were removed. The ipmap application then tells ovtopmd to make the appropriate changes to the topology database.

When netmon discovers a new node, ovtopmd adds the node to the IP topology database and informs ipmap that a new node has been discovered. The ipmap application uses what it knows about IP devices to tell ovw which icon and connection symbols it needs to create. The graphical interface then displays the correct symbols and modifies the map database accordingly.

For more information about the netmon and ovtopmd daemons, refer to “Background Processes” on page 8 or the man pages.

Map Synchronization

When a map that uses the ipmap application is opened, the application starts its synchronization phase. While the ipmap application is synchronizing, the graphical interface displays the [Synchronizing] message on the status line of all displayed submaps of the open map. During this phase, the ipmap application requests information about changes to the IP topology database since the map was last open. This information is continually updated by the ovtopmd daemon. If there are any new objects in the IP topology database, the ipmap application tells the ovw application which icons and connection symbols to add to the map.

The map will enter the synchronization phase for a short time during operations that change the contents of the map, for example, changing interface labels, adding new objects, or cutting and pasting large numbers of symbols.

Limitations: While the ipmap application is synchronizing, the following limitations exist:

- You cannot delete symbols, objects, or submaps.
- You cannot add objects.
- You cannot cut and paste symbols.
- The ipmap application will not appear in the Configurable Applications list on dialog boxes (Object Description).
- Manage Objects and Unmanage Objects operations will not take effect.
- Acknowledge and unacknowledge operations will not take effect.

When the synchronization phase completes, the ipmap application resumes full operation and the [Synchronizing] message is no longer displayed.

Using ipmap Application Submaps

The ipmap application and the graphical interface have rules that control the placement of symbols on submaps. Only certain symbols can exist on each submap type.

Table 7 on page 63 describes the submap hierarchy used by the ipmap application and which symbols are supported on each submap.

Table 7. Symbols Supported by ipmap

Submap	Symbols Supported
Root submap	IP Internet
Location or Internet	Location Internet IP network Connector (for example, a gateway) Connection
Network	Segment Connector (for example, a gateway) Connection
Segment	Node (for example, computers and connectors) Backbone Connection
Node	Interface card

Moving through the Submaps

Double-clicking on the IP Internet symbol takes you into the IP Internet submap, which is a special internet submap. The internet submap may contain a few network symbols. Double-clicking on a network symbol takes you into a network submap. On the network submap, you will probably have segment symbols. Double-clicking on a segment symbol takes you into a segment submap. The segment submap will have nodes connected to a backbone symbol. Double-clicking on the node symbol displays a node submap. The node submap displays all of the interfaces contained in the node.

Each time you select a symbol, you get a submap that is more specific than the previous submap. You can view your whole network, a part of the network, or a single node. Each time you view a submap's parent, you get a more expansive view of your network.

Root Submap: The root submap is the highest level submap. It is at the top of the Navigation Tree and has no parent. The root submap contains the IP Internet symbol created by the ipmap application. The IP Internet symbol is the only symbol on the root submap that the ipmap application manages. If you have other applications that use the graphical interface to draw symbols, you may have other symbols on the root submap, but ipmap does not know of their existence.

Internet and Location Submaps: Internet and Location submaps show logical groupings of IP networks and subnetworks connected by gateways. Location submaps and Internet submaps are considered to be equivalent by ipmap. This means that any symbol type that can be placed on a location submap can also be placed on an Internet submap.

The IP Internet submap is special because it is created by the ipmap application. It is also the highest level Internet submap, which means it is the only Internet submap that does not have a location or internet object as its parent. The root submap is its parent.

The ipmap application always places discovered IP addressable gateways and networks on this submap.

Internet and location objects are often referred to as container objects. That is because Internet and Location submaps are the only submaps that can contain their own symbol type. For example, a location submap can contain other location symbols, but a node submap cannot contain other node symbols. It can only contain interface symbols. This function can be used to organize and simplify a map. This is called partitioning. See “Partitioning Submaps” on page 74 for information about partitioning.

Network Submaps: The network submap represents the physical topology of a network at the level of network segments. The ipmap application can discover and display IP-addressable segments, gateways (routers), repeaters, multiports (hubs), bridges, and the connections between them on the network submap.

Note: If you are using IP subnetting, network implies subnet.

Segment Submaps: A segment submap represents the physical topology of a segment of your network at the level of nodes and connectors. It displays the computers and connectors that comprise a segment on your network.

In IP networks, a segment is a group of data communication objects that are interconnected through a common transmission medium. Nodes belonging to the same segment typically use a common physical medium to communicate with each other (for example, Ethernet, token ring, telephone lines, or satellite links). The following segment topologies can be drawn:

Bus	Represents nodes attached to a single linear cable that transmits data (for example, Ethernet or IEEE 802.3)
Token ring	Represents nodes attached to an SNMP, IP-addressable token ring central wiring MAU through twisted pair wiring, which conforms to the IEEE 802.5 standard
Star	Represents all nodes attached to an SNMP central multiport repeater (a hub)
FDDI ring	Represents nodes attached to an SNMP, IP-addressable Fiber Optic Data Distribution Interface (FDDI)

Node Submaps: The node submap displays symbols that represent the components of a node in a row/column layout. The graphical interface draws interface symbols on the node submap. When an interface object is added to a map, the ipmap application tells the graphical interface on which node submap to draw the interface card symbol. The ipmap application also tells the graphical interface on which higher level submaps to draw connection symbols.

Using the xxmap Application

The xxmap application enables you to see submaps showing information about open-topology objects. The xxmap application presents information gathered by the gtmd daemon. The gtmd daemon stores and correlates topology information received in the

form of SNMP traps or through API calls based on the NetView for AIX topology MIB. For more information about the gtmd daemon, refer to the *NetView for AIX Programmer's Guide* or the man page.

The xxmap application supports user changes to the map. Using the graphical interface, you can add and delete symbols. However, added symbols are not verified or stored in the gtm database. These symbols exist only in the user plane. When you delete a symbol from the map, the underlying object is not deleted from the gtm topology database. Objects can be deleted or added to the gtm database with an SNMP trap command or an API call. This symbol can also reappear during a subsequent map synchronization phase. If you do not want to see the symbol on a map, select **Edit..Hide Objects** to hide it. The object is still managed even though it is hidden. To view a list of objects that are hidden, select **Edit..Hidden Objects List**.

Displaying Multiple Protocols

If you are using protocols other than IP, the xxmap application enables you to see a list of the managed protocols running on a selected object. This option is available only for objects that are interfaces or nodes.

Suppose you select an object that is a computer running the CMIP protocol. Select **View..Protocols** to display a list that contains the following information about the CMIP protocol interface on the selected node:

- The name or address as defined in the NetView for AIX General Topology MIB
- The status of the protocol
- Submaps that contain the selected object's CMIP interface

Customizing a Graphical Map

There are many ways in which maps can be customized to make them easier to use and understand. Objects can be added to and deleted from your map database, and symbols can be moved from one submap to another. You can also group objects together into a collection based on a selection rule that you define, such as a collection of all routers.

Submap characteristics such as submap name, parent submap, layout, and background graphics can be changed. Also, objects can be selectively managed and acknowledged to match your management region responsibilities. The following sections describe how to use these customization functions.

Customizing Symbol Placement

You can organize symbols and submaps in ways that are easier for you. For example, you can add location symbols to your IP Internet Submap that correspond to your offices in New York, Chicago, and San Francisco. You can then move the network and gateway symbols from the IP Internet submap to the New York, Chicago, and San Francisco submaps, making network entities much easier to find. This type of customization is called submap partitioning.

Submap partitioning involves many different actions, including adding objects and moving symbols. Before we discuss submap partitioning, you should know the rules for adding, deleting, moving, and copying symbols and objects.

Note: A symbol can be an icon symbol, which is a two-dimensional picture, or a connection symbol, which is drawn as a line connecting two icon symbols. It is important to understand the difference between a connection symbol, which is a line that often represents an interface card object, and a connector symbol, which is a diamond shaped icon symbol that represents a device like a bridge or router. Throughout this section the term symbol refers to an icon symbol unless stated differently.

Adding Symbols and Objects

NetView for AIX allows you to add symbols to submaps. There are many reasons why you might want to do this. Here are two:

- A network entity may not have been automatically discovered by the netmon network discovery daemon.
- You want to add something to your map that is not really part of your network, like a location or segment symbol.

To add an object to your databases, you must know what object you wish to add, and on what submap you want the object's symbol to reside. Once you have done that, you simply drag the symbol from a palette of symbols to the correct submap, and then enter some information about the object. See “Steps for Adding Objects and Symbols” on page 68 for the steps on adding objects.

Note: Connection symbols are added differently from icon symbols. See “Adding Connection Symbols and Objects” on page 69 for information about adding connection symbols.

User Plane and Application Plane: A symbol that has a shadow is in the user plane, and a symbol that has no shadow is in the application plane. A symbol in the application plane is known and controlled by ipmap or some other application, whereas a symbol in the user plane is known and controlled by only the user that added or modified the symbol. Select **User Plane..For This Submap..Shadow Off** from the View pull-down menu to turn the shadow off an object in the user plane.

Symbols Managed by the IP Application: Certain applications only manage certain kinds of symbols. The ipmap application can manage the following types of symbols, organized by symbol class:

Table 8. Symbols Managed by ipmap

Symbol Class	Managed Symbols
Cards	IP interface card
Computer	All
Connector	All
Location	All
Network	Internet, IP Network, Bus, Token Ring, Star, FDDI, Serial, and Frame Relay segment

If you want the ipmap application to manage an IP entity, then use one of the symbols listed above. If you do not use one of the symbols listed above, then the symbols will not have status and connectivity information updated by the ipmap application.

Symbols That Can Be Added: The ipmap application enforces rules that control the placement of symbols on submaps. The ipmap application only manages symbols if they are placed on the correct submap. This ensures that submaps can always be traversed. It also ensures that connectivity and other relationships between objects can be conveyed properly.

Table 9 on page 68 describes symbols that can be added to the submaps and still be managed by the ipmap application. It is organized by submap type. The table also lists what information is needed about an object before it can be added.

Table 9. Symbols You Can Add That ipmap Can Manage

Submap	Symbols That Can Be Added
Root	None. The IP Internet symbol is the only symbol controlled by ipmap on this submap. Since this symbol is created for you, there is no reason to add any others.
Location or Internet submap	• IP network symbol (network class). You must provide unique network name, IP address, and subnet mask. • Gateway symbol (connector class). You must provide unique host name, IP address, and subnet mask. • All location symbols. You must provide unique selection name. • Internet symbol (network class). You must provide unique selection name.
Network submap	• All connector symbols. Provide a unique host name, IP address, and subnet mask. • Bus, star, token ring, FDDI, serial, frame relay symbols (network class). Provide a unique selection name.
Segment Submap	• All connector symbols. Provide a unique host name, IP address, and subnet mask. • All computer symbols. Provide a unique host name, IP address, and subnet mask.
Node Submap	IP interface card symbol. Provide an IP address and subnet mask.

Steps for Adding Objects and Symbols

Now that you know what and where symbols can be added, follow these steps to add an object and symbol. You must have the map open with read-write access to add objects.

Step 1. Select **Add** from the Edit pull-down menu

Step 2. Select **Object** from the Add pull-down menu.

Step 3. Select a class from the Symbol Classes on the Add Object Palette.

Step 4. Using mouse button 2, drag the desired symbol subclass icon to the appropriate submap and release the mouse button.

Step 5. When the Add Object dialog box is displayed, complete the following fields:

- Label
- Display Label
- Behavior
- Object Attributes
- Selection Name
- Comments

Step 6. If you want your added object to be updated, controlled, and managed by an application, then you have to:

   a. Select the application name from the Object Attributes list on the Add Object dialog box. If the symbol represents an IP entity, select the IP Map list item.

   b. Select the **Set Object Attributes** button.

   c. Fill in all the necessary fields.

Step 7. Select **OK** in the Add Object dialog box to add the symbol to the submap and close the dialog box.

Step 8. Select **OK** in the Add Object Palette to close the palette.

If this is done correctly, after you drag the symbol to the submap, you will see that the symbol is displayed flush against the background plane, and the symbol will be drawn without a shadow. The symbol is in the application plane. This indicates that the application you specified is correctly managing the symbol you added.

If you do not select an application and fill in the correct fields, the symbol is displayed raised above the background plane, and a shadow is displayed under the symbol. This means that the symbol is in the user plane and receives no updates from any existing applications.

The application that manages IP entities is called the ipmap application. When adding IP objects to your map, selecting the IP Map entry from the Object Attributes list and selecting the **Set Object Attributes** button ensures that your IP symbols will correctly reflect status and connectivity information.

Adding Connection Symbols and Objects

To draw a connection between two symbols, select **Edit..Add..Connection**. When you connect two IP symbols together, you are specifying that a new IP interface card exists that connects the symbols. For example, when you connect a gateway to a backbone symbol, you are adding an interface card to the gateway. The connection symbol is a graphical representation of the interface object in the object database.

Symbols That Can Be Connected: Table 10 on page 70 describes which IP symbols can be connected together with connection symbols. It is organized by submap type.

Table 10. Symbols That Can Be Connected

Submap	Symbols That Can Be Connected
Root	None
Location or Internet submap	Gateway symbol to IP network symbol
Network Submap	Gateway symbol to a bus, star, token ring, FDDI, serial, or frame relay segment
Segment Submap	All connector symbols to backbone symbol
Node Submap	None

Steps for Connecting Symbols

Now that you know what symbols can be connected, follow these steps to add a connection. You must have the map open with read-write access.

Step 1. Select **Add** from the Edit pull-down menu.

Step 2. Select **Connection** from the Add pull-down menu.

Step 3. Select one of the connection symbol types from the Add Connection palette.

Step 4. Select a source symbol on the submap that you want to connect.

Step 5. Select a destination symbol. The connection symbol is displayed between the two selected symbols on the map.

Step 6. When the Add Object dialog box is displayed, complete the following fields:

- Label
- Display Label
- Behavior
- Object Attributes
- Selection Name
- Comments

Step 7. If you want your connection symbol to be updated, controlled, and managed by an application, then you must:

a. Select the application name from the Object Attributes list on the Add Object dialog box. If the connection symbol represents an IP entity, select the IP Map option.

b. Select the **Set Object Attributes** button.

c. Complete the remaining fields.

If you don't do this, the connection symbol receives no updates from existing applications.

Step 8. Select **OK**.

Step 9. Select **OK** in the Add Object dialog box to set the connection in the map and close the dialog box.

Step 10. Select **OK** in the Add Connection Palette to close the palette.

The application that manages IP entities is the ipmap application. When connecting IP symbols on your map, select the IP Map entry from the Object Attributes list and select the **Set Object Attributes** button. This ensures that your connection symbols correctly reflect the status of the underlying interface card.

Deleting Objects and Symbols

To delete an object from the map, select **Edit..Delete Object..From All Submaps**. To delete an object completely from the object database, go into every map that contains that object and delete the object from all submaps. Use the Delete Object..From All Submaps option to be sure all symbols for an object are deleted from your map.

If the object is still part of your network, and if the netmon daemon is running, a deleted object can be discovered and displayed. There are two ways to prevent this:

- Turn off new node discovery by selecting **Options..Topology/Status Polling Interfaces: IP...** from the main menu.
- Let the object be discovered and then select **Edit..Hide Object** to hide the object. The object is still managed even though it is hidden. To view a list of objects that are hidden, select **Edit..Hidden Objects List**.

You must have a map open with read-write authorization to delete an object or symbol. You cannot retrieve deleted objects or symbols.

Moving Objects and Symbols

The Edit..Cut and the Edit..Paste functions are used to move symbols from one submap to another. The graphical interface lets you move any symbol you want to any submap you want. Although the graphical interface allows you to move any symbol you want, some symbols may be managed by particular applications that place limits on where symbols can be placed.

For example, the ipmap application only manages IP network symbols that are located on location or internet submaps. If an IP network symbol that is managed by ipmap is cut from a location submap and pasted onto a node level submap, then the ipmap application will stop managing that symbol. For this reason, it is a good idea to be familiar with the rules that each application places on the location of certain symbols.

As a general rule, if a symbol is in the application plane (the symbol has no shadow and is controlled by an application) and you move (cut and paste) the symbol, you want the symbol to end up in the application plane of the destination submap.

If you want the ipmap application to continue to control an IP symbol, then move (cut and paste) IP symbols to submaps that ipmap supports.

Symbols You Can Move: Table 11 on page 72 describes which IP symbols can be moved and still be managed by the ipmap application. They are listed by submap type.

Table 11. Symbols That Can Be Moved and Managed by ipmap

Submap	Symbols That Can Be Moved
Root Submap	None. The IP Internet symbol is the only symbol controlled by ipmap on this submap. This symbol must remain on the root submap.
Location or Internet submap	• IP network symbols (network class) • Gateway symbols (connector class) IP Network and gateway symbols can be moved from one Location or Internet submap to another Location or Internet submap. They should not be moved to any other type of submap.
Network submap	None. No symbols should be moved to or from a network submap. That is because the symbols on a Network submap have IP addresses and they would be inconsistent with the IP subnets of other networks.
Segment submap	• All connector symbols • All computer symbols All connector and computer symbols can be moved between segments in the same network. Node symbols should not be moved between segments in different networks.
Node Submap	None. No symbols should be moved to or from a Node submap.

When using the cut function to move symbols from one submap to another, always select the option Cut..From This Submap. Cutting from all submaps will cause multiple copies of your symbol to be moved, and that could cause links between symbols to be drawn incorrectly.

The cut and paste options allow you to move symbols from one submap to another. They should not be used to copy symbols.

Cut Buffer: The cut buffer holds symbols you have cut or copied until you do one of the following:

- Store another symbol into the cut buffer.
- Open another map.
- Exit the NetView for AIX program.

Steps For Moving an Object: Now that you know what can be moved and where it can be moved to, follow these steps to move an object. You must have the map open with read-write access to move objects.

Step 1. Open the submap that has the symbol(s) representing the objects you want to move.

Step 2. Select the symbols.

Step 3. In that submap, select **Edit..Cut..From This Submap**.

Step 4. Open the destination submap.

Step 5. Select **Edit..Paste**.

When symbols are pasted into a submap, they are first put in the user plane. Any applications that control or manage the symbols determine if they have been moved to a legal location. Depending on the machine speed and the number of symbols moved, this could take from 1 to 15 seconds. Once necessary applications determine that the move is supported, the symbols are moved into the application plane and the shadow goes away.

Copying Symbols

Like cutting and pasting, copying is a function that is provided by the graphical interface. Applications that use the graphical interface to control and display symbols provide various levels of support for the copy function.

The ipmap application does not support the copy function. When manipulating IP topology symbols, it is recommended that the Edit..Copy function not be used. If you want to move symbols between submaps, it is best to use the Edit..Cut function. IP symbols that are copied will be placed in the user plane rather than the application plane. These symbols will not show any connections to any other symbols. In general, the copy menu option on the graphical interface exists for use with other applications, and not with the ipmap application.

Steps For Copying Symbols: There are two parts to this procedure:

- Copying a symbol
- Pasting a symbol

Step 1. Copy the symbols.

Step a. Select the symbol that you want to copy.

Step b. Select **Copy** from the Edit pull-down menu.

Step c. Select one of the following from the Copy pull-down menu:

- From This Submap

  Performs the operation on this submap.
- From All Submaps

  Performs the operation on all submaps.

When you select Copy, the symbol remains on the submap.

Step 2. Paste the symbols.

Select the submap on which you want to paste the symbol. Select **Paste** from the Edit pull-down menu to paste the symbol onto the submap.

Note: You can paste the symbol onto any submap of the open map. Pasting the copied symbol onto the same submap from which you copied it creates multiple symbols of the object for the same submap.

All symbols are pasted in the New Object Holding Area for each submap in which automatic layout is disabled.

Partitioning Submaps

You can partition a submap to subdivide submaps into smaller, more managable units. For example, you can add location symbols that correspond to your offices in New York, Chicago, and San Francisco to your IP Internet Submap. You can then move the network and gateway symbols from the IP Internet submap to the New York, Chicago, and San Francisco submaps, making network entities much easier to find.

There are two kinds of submaps that you can partition:

- Internet and location
- Segment

The rules for each are slightly different.

Partitioning Internet/Location Submaps

Partition Internet or Location submaps when you want to take symbols from an Internet or Location submap and place them in other Internet or Location submaps. To partition an Internet or Location submap, follow these steps:

Step 1. Add either a location object or an internet object to either a location submap or Internet submap. If you are working with IP symbols, don't forget to fill out the correct IP Map information in the Add Object Dialog Box. See “Steps for Adding Objects and Symbols” on page 68 for steps on adding objects.

Step 2. Once the location or internet object has been properly added, select the network or gateway symbols that you wish to move into the new location.

Step 3. Select **Edit..Cut..From This Submap**. Once you have selected the symbols and cut them, they should disappear from the original submap.

Step 4. Open the submap of the internet or location symbol you added in the first step.

Step 5. Select **Edit..Paste**.

Example of a Partitioned Internet Submap

Figure 12 on page 75 shows an Internet submap in which four container objects were added. All symbols of objects discovered by the ipmap application have been cut from the Internet submap and placed in lower-level partitioned Internet submaps. If you clicked on the United States symbol, you would open the submap shown in Figure 13 on page 76.

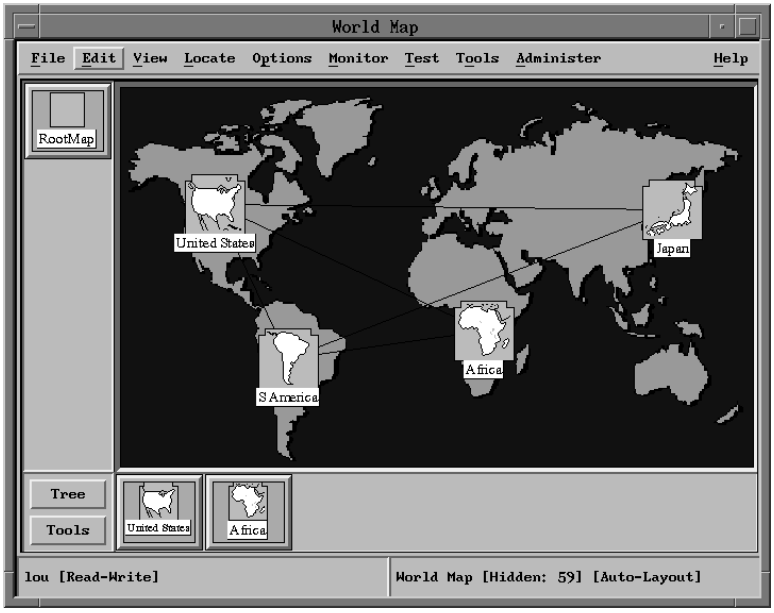


Figure 12. Example of a Customized Internet Submap

Figure 13 on page 76 shows the United States submap. Six container objects have been created in this submap. From this submap, you can double-click on any of the symbols and open a child submap to display other container objects or symbols that represent IP networks and gateways.

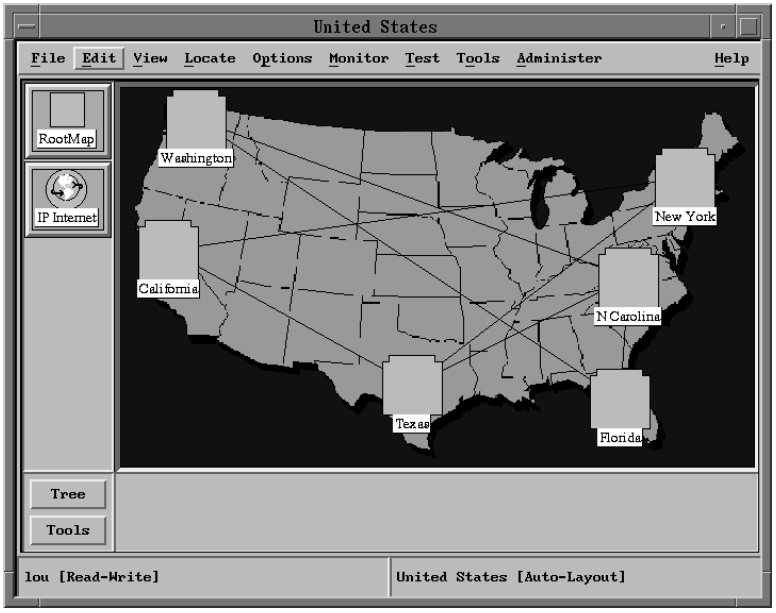


Figure 13. Example of a Partitioned Internet Submap

Partitioning Segment Submaps

Partition segment submaps when you want to move node symbols from one segment submap to another segment submap. To partition a segment submap, follow these steps:

Step 1. Open the network submap that contains the segment that you want to partition.

Step 2. Add a segment object to the Network submap. If you are working with IP symbols, don't forget to fill out the IP Map dialog box when adding the segment. See “Steps for Adding Objects and Symbols” on page 68 for steps on adding objects.

Step 3. Once the new segment has been added, open the segment submap that contains the symbols you wish to move.

Step 4. Select the symbols.

Step 5. Select **Edit..Cut..From This Submap**. When the cut is complete, the symbols should disappear.

Step 6. Open the new segment submap and select **Edit..Paste**.

Creating a Child Submap

Create a child submap if you need a more detailed view of an object. You can use the default settings or modified settings. Open the submap from any explodable symbol of the parent object. The submap is part of the map's hierarchy of submaps.

If the symbol you select already has a child submap, then select **Edit..Add..Object** to add an explodable object. This function adds a parent object from which you can add a child submap. Use the Location symbol type to represent the object if you are adding network views.

Using Default Settings

If you have a submap that consists of three groups of objects, organize your network by creating three child submaps, then select **Edit..Cut and Edit..Paste** to cut and paste each group of objects into a different child submap. Because you are only reorganizing your network, you do not need to change any of the default settings for the child submap.

Steps: To create a child submap with default settings, follow these steps. You must have the map open with read-write access.

Step 1. Double-click on an explodable symbol that does not contain a child submap. A NetView for AIX Windows Question box displays a message that the object does not have a child submap.

Step 2. To create a child submap with default settings, select the **OK** button. The new submap is created and displayed in a separate submap window. Because this is a new submap, the submap does not contain symbols or objects.

When you double-click on an explodable symbol of the parent object, the submap is opened. When you select **Show Parent** on the background context menu of the child submap, the parent submap is reopened in the separate submap window.

Default Settings: The default settings for the child submap are:

- The name of the submap is the same as the selection name of the object from which the submap was created.
- The presentation is scaling.
- The submap contains no background graphic.
- The submap contains no comments.

Using Modified Settings

If you have a submap that consists of three groups of objects that represent three parts of your network and you want to change the layout of one of the groups, you might consider creating three child submaps and changing the layout algorithm for one of the child submaps. After you create the child submaps and change the settings as necessary, use Edit..Cut and Edit..Paste to cut and paste each group of objects into the appropriate child submap.

Steps

To create a child submap with modified settings, follow these steps. You must have the map open with read-write access.

Step 1. Double-click on an explodable symbol that does not contain a child submap. A NetView for AIX Windows Question box is displayed, telling you that the object does not have a child submap.

Step 2. To create a child submap with modified settings, select **Modify**. The New Submap dialog box is displayed. The name of the new submap is displayed in the Name field. The name of the parent object is displayed in the Parent Object field. The button to the right of the Parent Submap displays the name of the parent submap.

Step 3. You can modify the following settings:

- Name
- Parent submap
- Layout
- Presentation
- Background graphics
- Comments

For more information about modifying these settings, see “Modifying Submap Settings” on page 79.

Step 4. After you complete the dialog box, select the **OK** button. The new submap is created and displayed. Because this is a new submap, the submap does not contain any symbols or objects.

Creating an Independent Submap

You can create an independent submap (also known as an orphan submap) that does not have associated parent objects. This type of submap is independent of the existing submap hierarchy. For example, you might create an independent submap that contains all your routers. This allows you to monitor router status from one submap.

To open an independent submap, select **View..Open Submap**. You can open a submap created without a parent object through the Navigation Tree window only if you have already opened that submap during the current NetView for AIX session. Use the horizontal scroll bar on the Navigation Tree window to see the independent submap symbols.

Steps

To create an independent submap, follow these steps. You must have the map open with read-write access.

Step 1. Select **Create Submap** from the Edit pull-down menu.

Step 2. A NetView for AIX Windows Question box is displayed. Select one of the following buttons:

- Select the **OK** button to create a submap with default settings. For more information about default settings, see “Creating a Child Submap” on page 76.

- Select the **Modify** button to create a submap with modified settings. For more information about modified settings, see “Using Modified Settings” on page 77.
- Select the **Cancel** button to cancel submap creation.

Note: Because this submap is an independent submap, you cannot assign it a parent submap.

Changing a Parent Submap

You can change the parent submap of the open submap. The parent submap is the submap whose icon is displayed last in the submap stack.

Steps

To change the parent of a submap, follow these steps. You must have the map open with read-write access.

Step 1. Select the submap for which you want to change the parent submap.

Step 2. Select **Modify/Describe** from the Edit pull-down menu.

Step 3. Select **Submap** from the Modify/Describe pull-down menu.

Step 4. The Submap Description dialog box is displayed. Select the Parent Submap option button to display a list of possible parents for this submap.

Step 5. Select a parent from the list. If no list is displayed, the current parent submap is the only choice at this time.

Step 6. After you select a parent submap from the list, select **OK** to apply the changes to the submap and close the dialog box.

Modifying Submap Settings

You can modify certain characteristics or information about a submap such as its presentation, the background graphic, or its parent.

Steps

To modify a submap, follow these steps. You must have the map open with read-write access.

Step 1. Decide which open submap you want to modify. Then make it the current window.

Step 2. Select **Modify/Describe** from the Edit pull-down menu.

Step 3. Select **Submap** from the Modify/Describe pull-down menu.

Step 4. The Submap Description dialog box is displayed. You can modify any of the following characteristics:

- Name
- Parent object
- Parent submap
- Presentation

- Background graphics
- Comments

Note: You cannot modify the submap layout. The layout algorithm can only be set during submap creation.

Step 5. After you modify the characteristics on the dialog box, select **OK** to apply the changes to the submap and close the dialog box.

Assigning a Home Submap

To assign the submap to be the home submap that will be displayed when the map is opened, follow these steps. You must have the map open with read-write access.

Step 1. Select **Set Home Submap** from the Options pull-down menu.

Step 2. Select the submap you want to be the Home Submap on the Submaps in Map dialog box. You can use the Find Submap field to locate a specific submap by matching a string or substring with a submap entry.

Step 3. Select the **Set as Home** button.

Step 4. Select the **Close** button in the Submaps in Map dialog box to apply the change and close the dialog box.

The selected submap is the submap displayed when this map is opened.

Modifying an Object Description

To modify an object description, follow these steps. You must have the map open with read-write access.

Step 1. Select one or more objects in the submap.

Step 2. Select **Modify/Describe** from the Edit pull-down menu.

Step 3. Select **Object** from the Modify/Describe pull-down menu.

Step 4. An Object Description dialog box is displayed for each selected object. You can modify the following attributes in the Object Description dialog box:

- Selection name
- Object attributes list
- Comments

Step 5. Select the **OK** button on the Object description dialog box to apply the changes and close the dialog box.

Managing and Unmanaging Objects

An object can be managed or unmanaged. A managed object is being monitored for topology, status, and configuration changes. The symbol for the managed object reports the status changes by changing to the color that represents the status. If an object is not being managed, the symbol for the object does not report the status because it is not known. However, the symbol is still visible. Managing objects uses network resources. If you have objects that don't need to be managed, unmanage them.

If you want an object to continue being managed but you don't want to see it, select **Hide Objects** from the Edit pull-down menu. The object receives and reports status but the symbol for the object does not appear on the submaps. You can also access the Hide operation by selecting **Edit..Hide Object** or **Edit..Hide..Symbol** from the object context menu. To view a list of objects that are hidden, select **Edit..Hidden Objects List**.

Steps

To manage or unmanage one or more objects, follow these steps. You must have the map open with read-write access.

Step 1. Select one or more objects to be managed or to be unmanaged.

Step 2. Select **Manage Objects** or **Unmanage Objects** from the Options pull-down menu.

All selected objects and child submaps are managed or unmanaged, depending on which option you selected.

Acknowledging and Unacknowledging Objects

If you know that an object has stopped functioning, but you do not want the NetView for AIX program to notify you continuously about this problem, use the acknowledge operation.

When you acknowledge the object, the object changes to a dark green color and remains in the acknowledged state until you select the object and unacknowledge it. Unacknowledging an object causes the NetView for AIX program to resume normal processing.

Steps

To acknowledge or unacknowledge one or more objects, follow these steps. You must have the map open with read-write access.

Step 1. Select one or more objects to be acknowledged or unacknowledged.

Step 2. Select **Acknowledge** or **Unacknowledge** from the Options pull-down menu.

All selected objects and child submaps are acknowledged or unacknowledged, depending on which option you selected.

Defining and Managing Collections of Objects

Network administrators often discover that, as their networks grow, distributing files and customization changes to nodes in the network is a time-consuming and error-prone task. The NetView for AIX collection facility provides a mechanism for you to group objects into a group. This group of objects is called a *collection*. You can use the included collection facility APIs to have your applications create and use collections.

The collmap application provides the display of collections and is automatically started when you start the graphical interface. You can use the Administer..Start

Application..collmap menu option to start the collmap application without having to close and restart the graphical interface. You might find this useful if the collmap application ends abnormally after you have started the graphical interface.

Note that collmap is a viewing tool only. It cannot be used to update collections in the collection facility. Thus, cut and paste are not supported within submaps.

Collections are aggregated under a Collection icon and are displayed on the Root submap. If you double-click on the Collection icon on the Root submap, you see a submap containing symbols for all the collections that are defined. Double-clicking on one of the collection symbols opens a submap containing all the objects that are currently in that collection. As objects move in and out of the collection, the submap is dynamically updated. A user in read-only mode needs to select **File..Refresh Map** to see new objects that have been dynamically updated.

Defining a collection can be useful for creating a submap of devices that you want to monitor closely. For example, you can define a collection of all critical routers. This enables you to view a submap of all routers that are inactive at any time.

You can collect MIB data for collections you have defined. See “Using the MIB Data Collector” on page 195 for information about how to collect MIB data.

If you have the IBM Systems Monitor program installed in your network, you can use the Agent Policy Manager to set thresholds and set up file monitoring for collections you have defined. For example, you can define a threshold to monitor CPU utilization for all objects in a Fileserver collection. The Agent Policy Manager is closely integrated with the collection facility. ***Management by policy*** in this manner facilitates your task of system management by centralizing control. If you need to change the thresholding on a group of objects, you do not have to make the change to each object in the collection; the change is applied automatically to all objects in the collection. Similarly, if an object that fits the defined collection is added to or taken out of the network, no additional changes are necessary. The collection facility automatically updates the collection.

See Chapter 8, “Using the Agent Policy Manager” on page 211 for more information.

Managing and Unmanaging Collections

The Collection icon and collection symbols are unmanaged until you double-click on the icons. Double-clicking on a collection symbol starts the monitoring process, and status is propagated upwards from objects within the collection. Because monitoring status uses network resources, you should unmanage collections that you do not need to be managed. To stop monitoring collection status, select **Edit..Delete..Symbol** to delete the collection symbol. Because you have not deleted the collection definition, the symbol is automatically recreated and is unmanaged. If you delete the collection icon on the IP root map, all the collections under the collection icon become unmanaged. If you want to unmanage a specific collection and continue to manage other collections, delete the collection symbol for the collection that you want to unmanage.

When you are managing a collection and new objects are discovered, the icons for these objects might appear as generic computers. This can happen if the ipmap application has not yet discovered the object or during map synchronization. Because the collection facility gets its information about which icon to create from the ipmap application and that information is not yet available, the collection facility uses the default icon, a generic computer. To create the correct icons, delete the collection symbol. The collection symbol will be recreated in the unmanaged state. When you double click on the collection symbol, the icons will be updated.

Types of Collections

To create a collection, you can specify the selection name of an object, or you can define a *rule* (much like a filter rule), using the NetView for AIX object capability definitions. If you specify a rule (such as `isRouter=True`), the collection facility locates objects that fit that description. A collection can also be a combination of a node list, rules, and other collections you have defined.

The Agent Policy Manager automatically creates one collection for you, and you will see it appearing on your root map with the label, MLM subnets. This special collection is made up of the mid-level managers (MLMs) in your network that NetView for AIX knows about, and all the objects managed by the MLMs. If you double-click on this symbol, NetView for AIX displays a map showing each of the MLMs, and double-clicking on an MLM symbol displays a star configuration of the MLM and the managed objects in its subnet. See “APM Collection Icons That You Get Automatically” on page 212 for more information.

A Quick Refresher Course on Boolean Logic

When you define collections, you have several opportunities to use logical AND and OR statements to join different rules you have specified. These logical operators have a different meaning than *and* and *or* in everyday speech. Consider this example:

- If I said to you, “Please bring me an apple and an orange,” you would return with two pieces of fruit, an apple and an orange.
- If I tell the collection facility, “Locate a mainframe computer AND a PC,” it will find nothing, because there probably is no individual device that has *both* of those characteristics assigned to it.

The logical operator OR also operates differently:

- If I said to you, “Please bring me an apple or an orange,” you would return with either an apple or an orange.
- If I tell the collection facility, “Locate a PC OR a mainframe,” it will find all the PCs and all the mainframes in the network.

When you use the collection facility, keep these simple rules in mind:

- When you want to find the **union** of two characteristics, use OR.
- When you want to find the **intersection** of two characteristics, use AND.

Adding a New Collection

To define a new collection, use the Collection Editor. You can use a list of IP addresses, capability rules, or a combination of both to define a collection. You can also use the name of another collection, thus building a hierarchy of collections.

The following example demonstrates how you can set up a collection for critical routers in your network. For this example, we will set up a collection that includes all of the Cisco and IBM routers. We will use the NetView for AIX predefined object capabilities to find objects that are classified as IBM routers or Cisco routers. We will add other routers by specifying their IP addresses. In addition, we will find those routers whose IP status is critical.

1. Collections are defined using the Collection Editor. Select **Tools...Collection Editor** from the NetView for AIX menu bar. The Collection Editor window is displayed.
2. Select the **Add** button to display the Add Collection dialog.
3. Enter the name of the collection, with a brief description of what the collection contains. We will call this collection `CriticalRouters`.
4. Define the first collection rule. Next to Definition 1, select the **Modify** button. The Modify Definition dialog box is displayed.
5. On this dialog box, select the **Definition Type** option button. You will see a list of the types of rules you can use to use to define a collection:

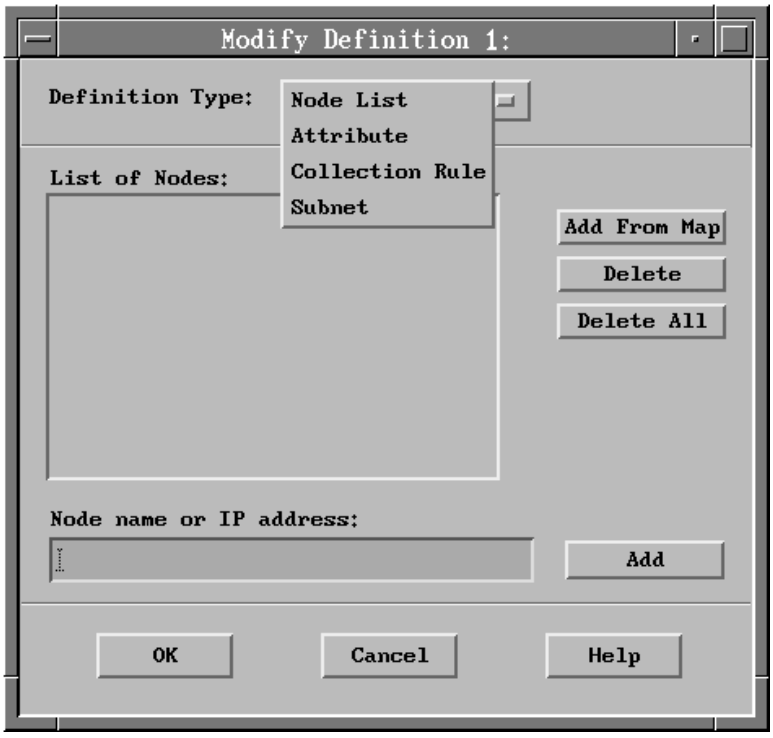


Figure 14. Collection Editor Modify Definition Dialog

6. Select **Attribute**.

   You will see a list of Object Attributes. Use the scroll bar to move down through the list until you find `vendor`.

7. Select the vendor object attribute. A list of selectable values is displayed:

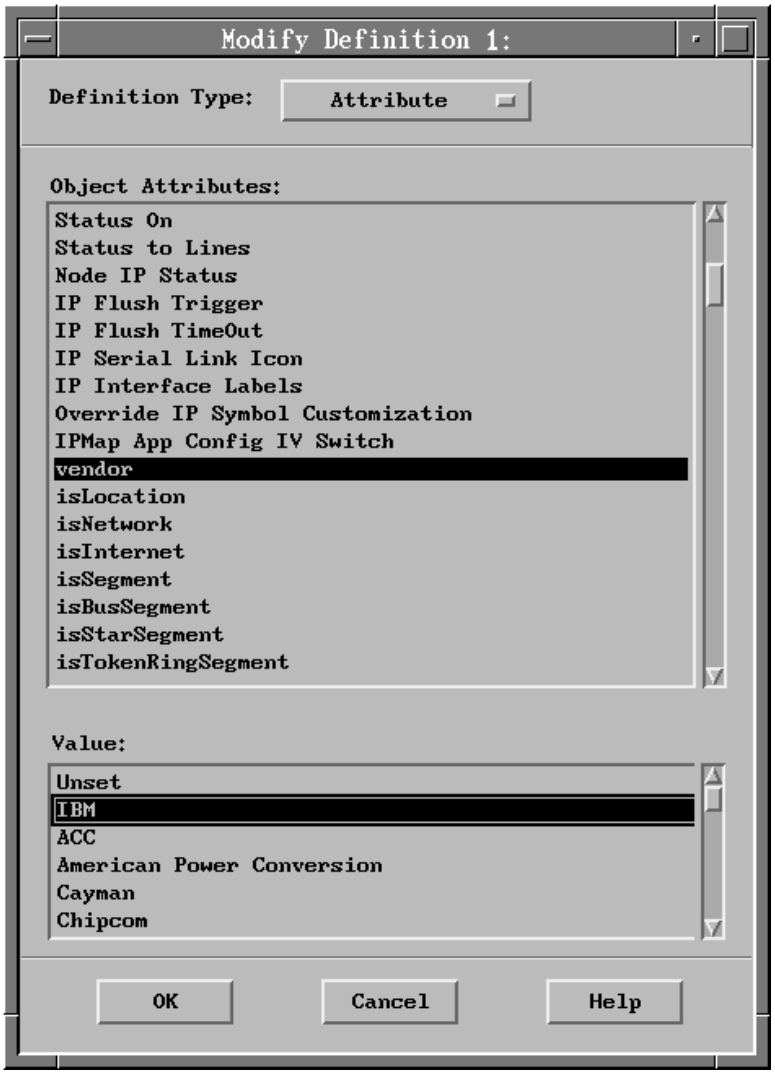


Figure 15. Collection Editor Object Attributes and Values

8. Select the IBM value and select the **OK** button. The definition is added to the Add Collection dialog box:

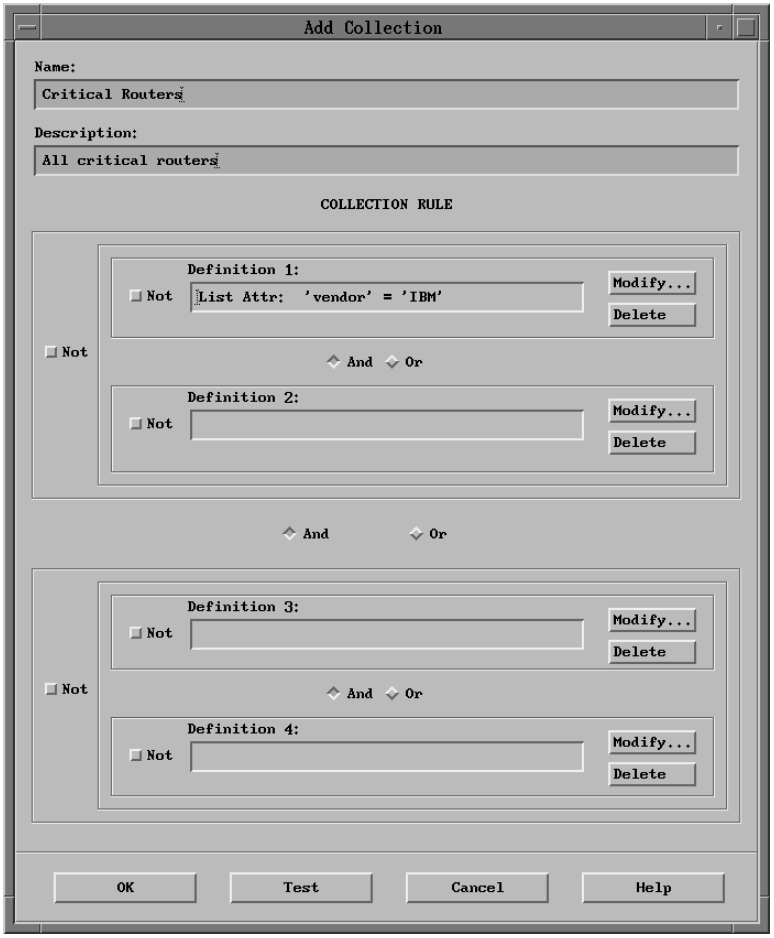


Figure 16. A Completed Collection Editor Definition

9. You now have a rule to select objects that have IBM specified as the value for their vendor attributes. To complete the first rule, you need to OR this rule with a rule selecting objects that have a vendor attribute of Cisco. (In other words, we will find objects that are made by IBM as well as objects that are made by Cisco.) Between Definition 1 and Definition 2, make sure the **Or** radio button is selected.
10. Select the **Modify** button next to Definition 2. Repeat the steps for selecting the vendor attribute, but this time select Cisco as the value for the attribute.
11. Select the **OK** button to add the definition.
12. You now have a rule to find IBM OR Cisco objects. You need to AND this rule with a rule that will find objects that are classified as routers.

    Make sure the **And** radio button between the top two definitions (1 and 2) and the bottom two definitions (3 and 4) on the Add Collection dialog box is selected.

13. Select the **Modify** button next to Definition 3. Select the Definition Type option button, and select Attribute from the menu that is displayed.
14. Scroll through the list until you find the attribute `isRouter`. Select this attribute.
15. This time, you will see that the attribute can be set to True or False. Select the **True** radio button and select **OK**.
16. You now have a rule to find IBM and Cisco routers. You need to AND this rule with a rule that will find routers that are down.

    Make sure the **And** radio button between Definition 3 and Definition 4 is selected.
17. Select the **Modify** button next to Definition type 4. Select the Definition Type option button, and select Attribute from the menu that is displayed.
18. Scroll through the list until you find the attribute `IP Status`. Select this attribute.
19. Select the Critical value and select the **OK** button.
20. You now have a rule to find all critical routers. Here is how the completed rule looks:

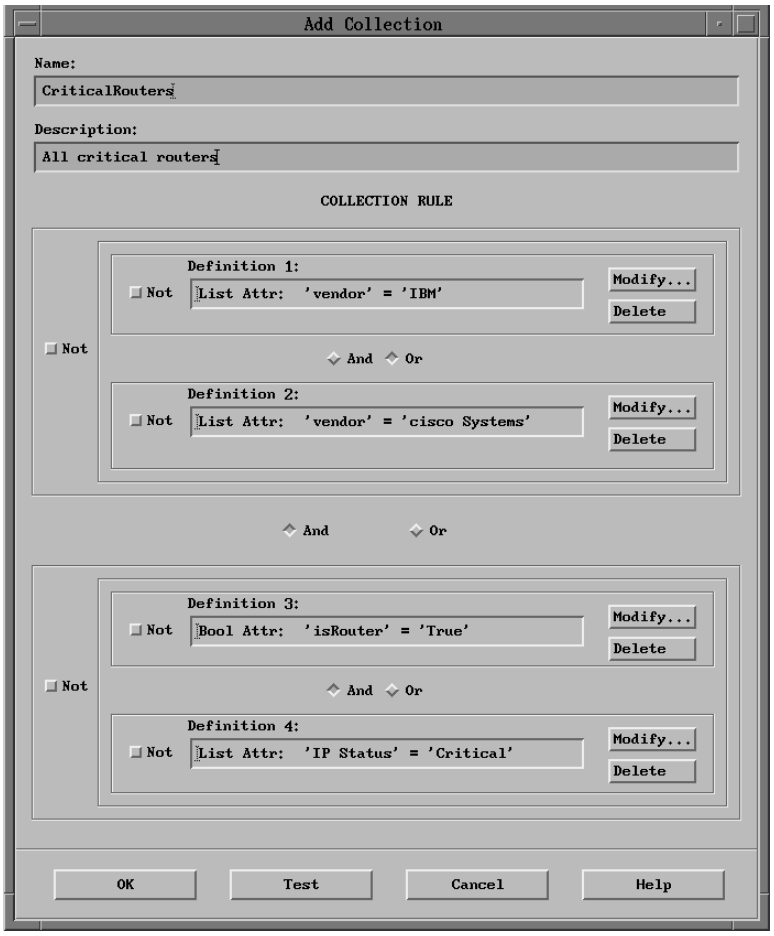


Figure 17. A Completed Collection Editor Rule

21. Select the **OK** button to add this definition. The collection you defined appears on the Collection Editor main window.

Modifying a Collection

Suppose you need to add one more router to your collection of critical routers. This router is a vendor that is not listed as a selectable value for the vendor attribute, but you do know the router's IP address. You can create another collection, using the CriticalRouters collection you already defined, and add the new router by specifying its selection name. Here are the steps:

1. Start the Collection Editor if it is not running. Select **Tools...Collection Editor** from the NetView for AIX menu bar. The Collection Editor window is displayed.
2. Select the **Add** button to display the Add Collection dialog box.

3. Enter a name and description for this new rule. We will call this example collection CriticalRoutersPlus1.
4. Next to Definition 1, select the **Modify** button. The Modify Definition dialog box is displayed. Select Include Collection Rule.
5. A list of defined collection rules is displayed. Select CriticalRouters and select the **OK** button. The collection name is displayed in Definition 1.
6. Next to Definition 2, select the **Modify** button. The Modify Definition dialog box is displayed. This time, we will use Node List as the Definition Type. It is probably already selected; if not, select Node List.
7. At the bottom of the Modify Definition window, enter the selection name of the object to be added in the Node Name or IP address field and select the **Add** button. The object is added to the List of Nodes.
8. Select the **OK** button. The object is displayed in Definition 2.
9. Select the **And** radio button between Definition 1 and 2 and Definitions 3 and 4.
10. Next to Definition 3, select the **Modify** button. The Modify Definition dialog box is displayed. Select Attribute as the Definition Type.
11. Select the attribute `IP Status` and the value `Critical`.
12. Select the **OK** button. The IP status is displayed in Definition 3.
13. Select the **OK** button to add this definition. The collection you defined appears on the Collection Editor main window.

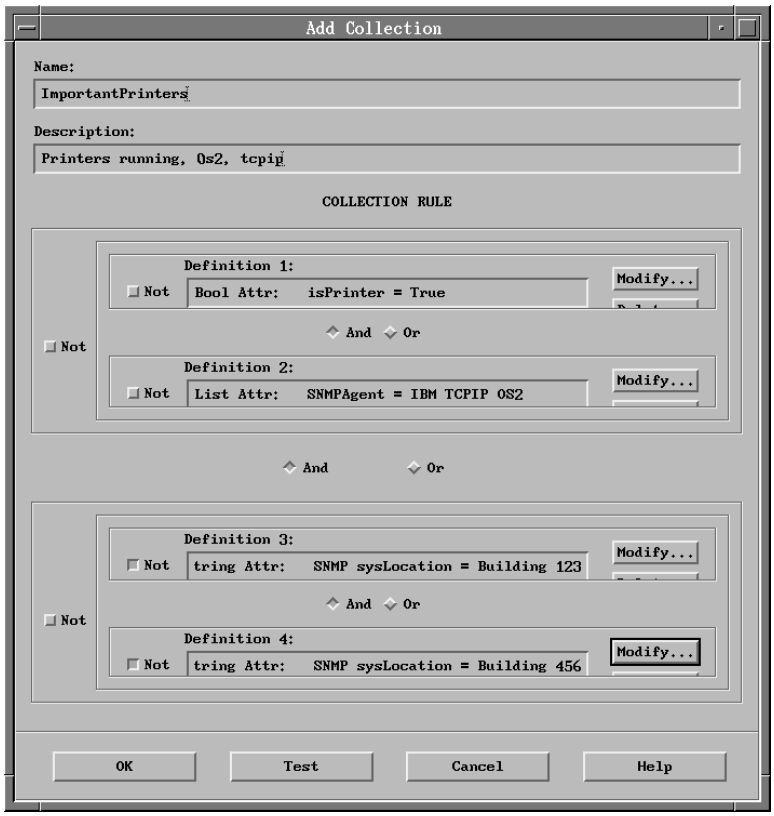
Listing Objects in a Collection

To find out which objects in the network are included in a collection, select a collection on the Collection Editor window and then select the **Resolve** button. A list of objects in the selected collection is displayed.

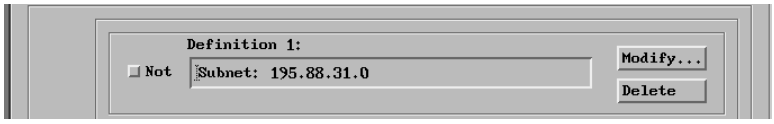
More Examples of Collections

Here are a few more examples of collections you might develop. Each example has a statement of the rules under which an object is to be included in the collection and a picture of how the rule is to be defined in the collection facility.

Example 1: Make a collection of all printers running OS/2 TCP/IP SNMP agents that are not located in building 123 or building 456.



Example 2: Make a collection of all nodes contained in subnet 195.88.31.0.



Chapter 4. Customizing the Graphical Interface

Customizing the graphical interface can mean as little as adding backgrounds or it can include specifying maps for startup, authorizing file permissions, setting event filters, and customizing menu bars.

This chapter describes the tasks involved in customizing the graphical interface. You can change the graphical interface for each user so they see only what they are responsible for. For example, if you have three users, each responsible for a different geographical region, you can create or customize submaps for each user so they see only the region for which they are responsible. You can give them access to applications that pertain to only them. If you want to add a different background to each submap, you can do that also.

This chapter includes the following tasks:

Adding or Removing a Background Graphic

You can add a graphic to the background of a submap or replace a graphic with a different graphic. Background graphics can provide contextual information, such as:

- A floor plan of your business
- A geographic map showing diverse sites
- A diagram representing some characteristics of a portion of the managed network

Supported Formats

The graphical interface supports the following file formats for background graphics:

GIF**	CompuServe** Graphics Interchange Format Background graphics must be in GIF87a format.
XBM	X11 monochrome bitmap format

Adding a Background

To add a background graphic or replace an existing background graphic of a submap, follow these steps. The map must be opened with read-write access.

Step 1. Choose **Select Background Picture** from the Edit pull-down menu.

The graphical interface displays the Submap Description dialog box for the open submap.

Step 2. Complete or change the Background Graphics field by entering a line with the complete path and name of the graphic file. For example, to select the background graphic for a map of the United States, enter the following:

```
/usr/OV/backgrounds/usa.gif
```

Or, click on the **Select** button and select a graphic from the list that is displayed.

Step 3. To display the new background graphic, select the **OK** button.

If you do not select a background graphic, the background plane remains empty.

Removing a Background

To remove a background graphic, delete the input file name from the Background Graphics field. The full path name must be deleted. The map must be opened with read-write access.

Step 1. Choose **Select Background Picture** from the Edit pull-down menu.

The Submap Description dialog box for the open submap is displayed.

Step 2. Select the Background Graphic field to make it the active field. Make sure the cursor is at the beginning of the text.

Step 3. Press Ctrl and then Delete.

The graphic name is removed and the field is blank.

Step 4. Select **OK** to close the Submap Description dialog box.

Arranging Symbols

You can arrange the symbols that are displayed on the submaps either automatically or manually. Whichever method you choose, you must have a read-write map to save the new layout. You can also change the symbol labels. This section provides the steps for arranging and labeling symbols.

Using Redo Layout

Redo layout enables you to arrange the symbols in a submap according to the assigned layout algorithm. Redo layout works regardless of whether automatic layout is enabled or disabled.

To save the new layout for the submap, you must have read-write access. New symbol positions are not saved for read-only maps.

Steps

To redo the layout for a submap, follow these steps:

Step 1. Select **Redo Layout** from the View pull-down menu.

Step 2. If you redo the layout for a read-only map, the graphical interface displays a warning message that the new symbol positions will not be saved. If you redo the layout for a read-write map, the graphical interface displays a warning message that all symbols will be repositioned and their current positions cannot be restored.

Step 3. Select the **OK** button to redo the map. The graphical interface repositions the symbols on the submap according to the submap layout algorithm.

Setting Automatic Layout

Automatic layout enables the system to automatically arrange the symbols in a submap according to the assigned layout algorithm. You can turn automatic layout on or off for the current submap, or for all submaps. You must have the map open with read-write access.

Steps

To set automatic layout, follow these steps:

Step 1. Select **Automatic Layout** from the View pull-down menu.

Step 2. The graphical interface displays a cascade menu. You can turn automatic layout on or off for the current submap or for all submaps.

Step 3. After you make your selection, the graphical interface applies the change.

Note: If you turn automatic layout off, for either the current submap or for all submaps, the graphical interface requires a holding area to place the newly discovered objects. The graphical interface displays an area at the bottom of the current submap or all submaps called the New Object Holding Area. The New Object Holding Area is displayed only when there are symbols to be placed.

To manually move an object from the New Object Holding Area, select the object using the Ctrl button and mouse button 2 and drag it to the current submap.

To enable the graphical interface to move all objects from the New Object Holding Area to the current map, select **View..Redo Layout** or turn automatic layout on.

Modifying and Displaying Symbol Labels

If you have changed the name of a resource and you want to change the label for the symbol, follow the procedure to modify and display symbol labels.

Steps

To modify symbol labels, follow these steps. You must have the map open with read-write access.

Step 1. Select the symbol of the object with mouse button 3 to display the context menu.

Step 2. Select **Modify/Describe** from the Edit pull-down menu.

Step 3. Select **Symbol** from the Modify/Describe pull-down menu.

Step 4. Enter the text in the Label field on the Symbol Description dialog box.

Step 5. Select **Yes** or **No** to the right of Display Label to select whether to display the symbol label in the submap.

Step 6. Select the **OK** button on the Symbol Description dialog box to apply the change and close the dialog box.

Note: If there are too many symbols on a submap, the labels are not displayed.

To control the display of selected symbols, select **Show/Hide Labels** from the Edit pull-down menu.

If you want to change the symbol on a submap, select **Change Symbol Type** from the Edit context menu.

Assigning Maps

You can provide read-write access for each user by creating a unique map for each user. The map can be a copy of the map named default. Users can make changes to their own map without effecting what is displayed on other maps. When you create a map, you become the owner and the only one with read-write access to the map. Use the **ovwchown** command to change the owner of the map from you to the user.

Setting Map Permissions

To prevent maps from being deleted, set up different map permissions on the various map databases.

Use the **ovwperms** command or the SMIT option Configure..Change Maps to set permissions on maps. Setting map permissions to read-only for users prevents them from deleting specific maps. Only the root user can then delete a map. You must be a root user to change map permissions. See “Assigning Map Access Levels” on page 53 for information about the different types of map access.

The **ovwperms** command changes the permissions for all files and directories associated with the map. See the ovwperms man page for more information about changing map permissions with a command.

Steps

To change map permissions using SMIT, follow these steps:

Step 1. Access SMIT by entering **smit nv6000** on the command line or selecting **NetView SMIT** from the Administer pull-down menu.

Step 2. Select **Configure**.

Step 3. Select **Change Map(s) owner/group/mode**.

Step 4. Enter the necessary information in the fields.

Step 5. Select **Do**.

Step 6. Exit from SMIT.

If you have a number of users that share the same maps and applications or perform similar tasks, use SMIT to create a group for them. Groups can be formed for users that share access authority to protected resources. Once your group is established, you can change map permissions for the group rather than for individuals.

Customizing the Menu Bar and Tool Palette

If you are not using NetView for AIX security services, which provides a customized graphical interface based on NetView for AIX group permissions, you can use the OVwRegDir environment variable to customize each user's menu bar and tool palette with the applications that pertain to only that user. See Chapter 2, “Defining and Managing a Security Policy” on page 19 for information about NetView for AIX security services. The OVwRegDir environment variable points to the location of the application registration files. You can set this variable in the user's .profile or .kshrc file. Placing the desired registration files in each user's directory gives each user access to a different set of operations. You can point to individual directories, then from those directories, set up symbolic links to the registration files you want in /usr/OV/registration/C.

If there are specific menu items you do not want users to access, edit the ovw application registration file to remove those entries. However, doing so prevents anyone from using those options because they no longer exist.

An alternative to deleting menu items is to split the ovw application registration file. For example, you can split the file in two, ovw1 and ovw2, and separate the options. Add the OVwRegDir environment variable to the user's .profile to point to either the ovw1 or ovw2 registration file.

For example, let's say we split the ovw file into ovw1 and ovw2. You have two users, Lou and Judy. If you want Lou to have access to the menu items in ovw1 and Judy to have access to the menu items in ovw2, add the following line to each of their .profiles:

```
export OVwRegDir=/u/lou/reg/C          For Lou
export OVwRegDir=/u/judy/reg/C         For Judy
```

Copy ovw1 to Lou's directory:

```
/u/lou/reg/C/ovw1
```

Copy ovw2 to Judy's directory:

```
/u/judy/reg/C/ovw2
```

Before editing the ovw registration file, make a copy of the original for a backup file.

Changing the Graphical Interface Defaults

There are certain graphical interface characteristics that you might want to change. For example, you might want the navigation tree and tool palette to display as icons, or you might not want them to be active at all when NetView for AIX is started. The resources and defaults for these characteristics are defined in the /usr/OV/app-defaults directory for all users.

You can change a default for each user by copying the line containing the resource you want to change into the users $HOME/.Xdefaults file. Changing the resource in the app-defaults files affects all users. You might prefer to put the entries in the .Xdefaults file because customized settings in the app-defaults files are overwritten if you apply a service fix, but the settings in the .Xdefaults file are not. Settings in the .Xdefaults file override the settings in the app-defaults files.

If you make changes to any of the app-defaults files after NetView for AIX is started, use the command `xrdb -merge .Xdefaults` to load the new resource. If you have not started NetView for AIX, the new resource will be loaded when you start the graphical interface.

To change the files in the /usr/OV/app-defaults directory, you must be a root user.

Window Resources

Resources are found in the /usr/OV/app-defaults/OVw file, which defines the resources for the NetView Windows server. Although there are too many to list them all here, if you browse the file, you'll find resources for the following:

Fonts	You can change all fonts in the graphical interface, such as button fonts, label fonts, and window title fonts.
Colors	You can change all colors in the graphical interface, such as background colors, symbol colors, and connection colors.
Sizes	You can change the default sizes for all windows in the graphical interface, such as message windows, submap windows, and the tools window.

Other Resources

Table 12 on page 99 lists files, resources, and defaults found in the /usr/OV/app-defaults directory. This is not a complete list. Browse the files to see a complete list. These are resources that affect the appearance of applications displayed in the graphical interface. If you are changing the resources for an individual, you must have read-write access to the .Xdefaults file in the $HOME directory you are changing. If the .Xdefaults file does not exist, you can create this file.

Table 12. Xdefault Resources

What You Want Changed	File	Resource	Default
Event card format presentation	Nvevents	nvevents.initialPresCard	True
Event cards text color	Nvevents	nvevents*card*cardTextColor	black
Event cards color	Nvevents	nvevents.cardColor	#ffdcedf3d2d2
Event application text color (text that doesn't appear on the cards)	Nvevents	nvevents*foreground	black
Number of events in a workspace	Nvevents	nvevents.maxLoadEvents	500
Number of workspaces per session	Nvevents	nvevents.maxNumWS	500
Save event workspaces	Nvevents	saveEnvOnExit	False
Start event application with saved workspaces	Nvevents	loadEnvOnInit	False
Include static workspaces and workspaces that were loaded using the File..Load option for saving	Nvevents	considerStatisWrkSpcs	False
Start main window and control desk as icon	OVw	OVw*shellIconify	False
Start the tool palette as icon	OVw	OVw*toolShellIconify	False
Start the navigation tree as icon	OVw	Ovw*navTreeShellIconify	False
Start the navigation tree	OVw	OVw*navTreePresent	True
Start the tool palette	OVw	OVw*toolPalettePresent	True
Start control desk with events minimized	OVw	Ovw*thereAreInternalTools	True
Start contol desk with network view	OVw	OVw*controlDeskHasBox	True
Graph line width	XNm	xnmgraph.lineWidth	2
Graph line colors	XNm	xnmgraph.graphLineColors	See file

Steps for Changing the Graphical Interface Defaults

To customize the graphical interface defaults for a user ID, follow these steps:

Step 1. Examine the files in the /usr/OV/app-defaults directory to determine which items you want to customize.

Step 2. Edit the .Xdefaults file in the user's $HOME directory and implement your changes. For example, if you want to change the default font for the Tree button on a submap window, enter a line with the following format:

```
OVw*treeButtonFont:              <fontname>
```

where <fontname> is the name of the font that you want. For more information about fonts, colors, and sizes, refer to the AIXwindows* product documentation.

Step 3. After you change the .Xdefaults file, save it. To see the changes, you must exit the NetView for AIX program and restart it or use the command `xrdb -merge .Xdefaults` to load the new resource.

Preventing the Control Desk from Automatically Starting

By default, the Control Desk is active when you start the graphical interface. If you don't want the Control Desk to start, edit the /usr/OV/registration/C/ovsnmp/nvevents file, and remove the –`Initial` flag from the following line:

```
Command -Shared -Initial "$[nvevents:-/usr/OV/bin/nvevents]" ;
```

Note that the Control Desk will start by running any application, such as the Event Display application, that uses the Control Desk, and you can start the Control Desk from the tool palette. When you apply NetView for AIX service, a new /usr/OV/registration/C/ovsnmp/nvevents might replace your customized file, and you might need to remove the -Initial flag from the new file.

Customizing the Failing Resource Display

If you have a read-write map you can display all the failing resources for an object by selecting **Tools..Failing Resources Display**. This option opens a new submap with symbols that represent the failing resources of a selected object. This new submap is not connected to the submap hierarchy for the currently opened map. The failing resource submap starts its own hierarchy which you can see in the Navigation Tree.

The failing resource symbols reflect the current status of all the interfaces of an object. If an interface status changes, the symbol representing that interface also changes. When you double-click on a failed resource, another submap displays the hierarchy of the failed resource within the network.

You can customize the Failing Resource Display by editing the /usr/OV/registration/C/xnmfault file and setting the following flags:

-t Changes the text in the title bar.

-r Changes the capability field so that only certain objects are displayed. For example, you might want to display only those objects that have the field isConnector set to True.

-s Changes what the status of the resources must be to be displayed. For example, you can change the status so that only marginal status is displayed.

-u 0|1
A zero (0) indicates that if symbols matching the criteria are not found, continue checking and display a submap when one is found. A one (1) removes the symbols if the status does not match the criteria.

You must restart ovw for changes to take effect. The following is an example of the command line in the xnmfault file:

```
Command "/usr/OV/bin/xnmfault -t 'title' -r isConnector -s 2 \"$OVw
    Selection1\"";
```

Customizing Event Filters for Users

You can start the nvevents application with activated filters. During startup, the nvevents application reads an event filter file located in the user's $HOME directory or in the directory indicated by the resource `profileDir` in the /usr/OV/app-defaults/Nvevents file. The filter file is named after the user with the extension of `events`. For example, if the user name is shannon, and you did not modify the .Xdefaults file, the filters are defined in $HOME/.shannon.events.

You can define which filters you want activated for each user. The events file name and the filter rule name must be defined in the user's events file using the following syntax:

```
FilterFileName filterFileRule
```

For example, if you want to activate filters Trap_to_Alert_Threshold and Receive_from_6611_router, then the contents of the events file will look like the following:

```
/usr/OV/filters/filter.samples Trap_to_Alert_Threshold
/usr/OV/filters/filter.samples Receive_from_6611_router
```

Where `filter.samples` is the filter file name and `Trap_to_Alert_Threshold` is the filter file rule.

When you activate filters during the nvevents operation, they are saved in this file. The next time the application is started the last filter(s) activated will be automatically registered. See Chapter 5, “Correlating, Filtering, and Configuring Events” on page 109 for more information about filtering events.

Customizing the NetView for AIX Grapher

Many of the NetView for AIX program's graph applications present their results in graphs that you can save, print, or customize to better suit your needs. Although each application graphs different information, the general presentation format is the same.

Graph applications have certain display default values. These defaults might be suitable for your purposes, but there might be other times when you need to modify the default display to meet your needs. This section describes the ways you can customize the graph display.

Figure 18 shows the graphical interface's standard layout for graph applications, using the Monitor..Network Activity..Interface Traffic application.

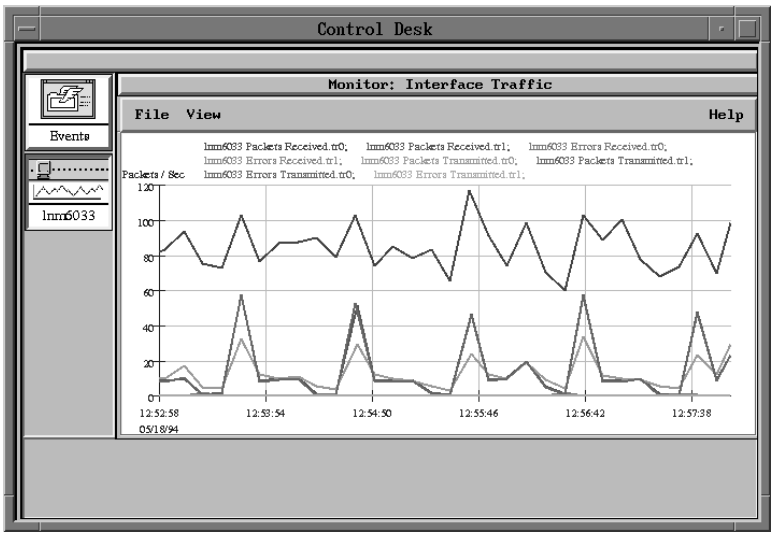


Figure 18. The NetView for AIX Grapher's Graphical Interface

This time, the application was started from the NetView for AIX main menu, so it is displayed in the control desk. You might choose to start it from the Tools window and not put it in the control desk.

Entering Numeric Values

The following rules apply when entering numeric values in dialog box fields:

- A number beginning with x or 0x is treated as hexadecimal. For example, 0x0010 becomes decimal 6.
- Any other number beginning with 0 is treated as octal. For example, 010 becomes decimal 8.
- Any number beginning with a numeric digit other than 0 is treated as decimal. For example, 10 becomes decimal 10.

- If you enter a value containing a digit that is not valid for its format, the number is truncated at the place that was not valid. For example, 123A56 becomes 123.

Setting Time Intervals

Select **View..Time Intervals** from the menu bar to see the time statistics associated with the graph application. The Display Interval at the top of the dialog box shows the date and time the application was started and is dynamically updated with the current date and time. Move the slider box below the Display Interval to change the interval for which graphed data is displayed.

By default, the graph display shows the most recent 5 minutes of results. The vertical bars on the display mark the minutes. You can change the display width by typing a new interval in the Display Width text field and selecting the **Apply** button. Notice that a beginning date and time and an ending date and time are displayed directly above the Display Width text field.

You can change the resolution of the graph by selecting the Resolution by data / Resolution user defined option button. Resolution by data means that the graph displays the data exactly as it is collected. If you select Resolution user defined, you can increase the resolution, which normalizes the data and enables you to see trends. Refer to the Help system for more information about changing the graph resolution.

The **SNMP Polling On / SNMP Polling Off** option button enables you to set the frequency with which a graph of real-time data is updated. This button has no effect on historical data. The update frequency determines how often the device is queried and any new data displayed in the graph.

Changing the Line Configuration

You can change the characteristics of the lines on the graph. Select **View..Line Configuration** to display the Line Configuration dialog box, as shown in Figure 19 on page 104.

Figure 19. The Line Configuration Dialog Box

The Line Configuration dialog box displays a row of fields for each line on the graph. The fields of the Line Configuration dialog box are described in the following list.

Data Label
: This section lists the names of all MIB objects for which the application is collecting and graphing information. By selecting a name from the list that is displayed when you select the option button, you can choose which MIB objects you want to display. This feature is useful when an application gathers data on more MIB objects than the maximum number that can be displayed at one time in the graph.

On/Off
: If you have many lines on your graph and want to simplify the presentation, turn some lines off by clicking the **On/Off** toggle button. You can also turn lines off by selecting the lines on the graph with the Ctrl button and mouse button 2. The lines' data labels are also removed from the display. To turn the lines back on, click the **On/Off** toggle button to On.

Color
: You can change the color of each line by selecting a color from the list that is displayed when you select the Color toggle buttons. If it is difficult for you to distinguish certain colors and, therefore, differentiate the lines, you can display the label for the line by selecting the line on the graph with mouse button 2.

Line Width
: You can change the line width by selecting a line width from the selections that appear when you select the **Line Width** toggle button.

Multipliers — Sometimes the collected data includes values that are reported in different units of measurement. You can display all of them on one graph for comparison by changing multipliers, so they are all based on the same unit of measurement. The default is to display data with no multiplier.

Displayed Values — By default, the average values are displayed. You can select any of the following values for display:

- Minimum
- Average
- Minimum and Average
- Maximum
- Minimum and Maximum
- Average and Maximum
- Minimum, Average, and Maximum

For example, if you select Minimum and Average, the graph will display two lines for the same MIB object, one that reflects the Minimum value and one reflecting the Average value.

Getting Application Statistics

Select **View..Statistics** from the menu bar in the graph application to see traffic statistics about each line on the graph. Information about the minimum, average, maximum, and last values is displayed in a table where each line of graphed data forms a row of the table. The statistics are updated based on the value specified for SNMP polling intervals. These statistics are used for analyzing trends about performance peaks and valleys.

By default, the raw values are shown. You can select a multiplier to change the y-axis increments. Doing so also changes the shape of the lines on the graph. To change the multiplier, select **View..Line Configuration**. Choose a value from the list that is displayed when you select the **Multiplier** option button.

Checking Application Messages

You can look at all the messages that have accumulated for the graph application by selecting **View..Messages** from the menu bar. The Messages dialog box displays error and informational messages associated with the application's processing. Output from the File..Memory Usage and File..Line Info operations is also stored in this dialog box.

Paging through the Graph

If you are collecting data over a long period of time, the graph cannot display all of it at once. Select **View..Screen Paging..** for a Help dialog box that explains how to view different parts of the collected data. You can page backward or forward, one screen at a time, or center the graph around a selected point in time.

Scaling the Y-Axis

You can scale the y-axis of the graph in one of the following ways:

- On all data
- On displayed data

The default display scales the y-axis on displayed data. You can use the default if you are not running the application for a long period of time. However, if you run the application for a long time, there might be a greater fluctuation from high to low values, which might not be reflected in the portion of the graph currently being displayed. If you need to base subsequent decisions on the overall pattern of data, consider changing the scaling.

Changing the Display from Color to Monochrome

The default graph presentation uses a different color for each line that represents a monitored MIB object. The colors help you track which line goes with which MIB object.

You can change to monochrome mode and still differentiate each line on the graph. Select **View..Color/Monochrome** from the menu bar and then select **Monochrome**. Monochrome mode uses different types of lines, for example, solid or dotted, to represent each MIB object's values. You might want to use this option to see how the graph would look if it were to be printed on a monochrome printer. You can use the Print Tool to see the results.

Displaying or Hiding the Grid

By default, a grid is displayed for all graph applications. You can choose to hide the grid by selecting **View..Show/Hide Grid** from the menu bar. The x- and y-axes remain, but the vertical lines that mark each minute disappear, as do the horizontal lines that extend the numerical divisions on the y-axis across the width of the graph.

Showing Counter Values

If you are graphing MIB values of type Counter, you can select **View..Show counters As...** from the menu bar and choose one of the following ways to display the data:

Rate of Change	The default value, which shows the new counter value as a time-averaged value since the last query for the MIB object.
Actual Sampled Value	Shows the actual value returned from the MIB Counter variable.
Delta Value	Shows the actual change in the MIB variable since the last query for the MIB object. This value is not time-averaged.

For example, suppose you are graphing a MIB variable with the following statistics:

```
Value of MIB variable at time 0  --> 100
Value of MIB variable at time 10 --> 300
```

The value from `time 10` would be graphed in the following ways:

Rate of Change	20 (derived from (300–100)/10)
Actual Sampled Value	300
Delta Value	200 (derived from (300–100))

Adding a Line

An application sometimes has more lines to graph than the maximum number that are specified for the graph when the application is created. You can use the View..Add Line operation to temporarily increase the number of lines the graph can display. This selection will be grayed if the number of lines to be graphed does not exceed the maximum number of lines that can be graphed, as defined in the application's app-defaults file.

Using the Context Menu

The context menu in a graph application enables you to zoom in and out so you can look at the graph from different perspectives. You can also use the context menu to page forward or backward through the collected data, or to display the beginning of the data, the end of the data, or all of the data.

Printing Graphs

You can use the Print Tool application to print graphs. See “Printing Graphed Data” on page 206 for these steps.

Chapter 5. Correlating, Filtering, and Configuring Events

To help you manage a network effectively, the NetView for AIX program must receive information about changes that affect objects in the network. *Events* generated by agents that monitor network objects convey this information to the NetView for AIX program.

Large networks with many objects and agents can generate so many events that the manager is flooded with traffic and must devote an excessive amount of time to processing incoming events. In addition, the manager generates events when it polls agents for the status of network objects. *Event correlation rules* and *event filters* can help you control the amount of event traffic to be displayed on the NetView for AIX graphical interface or forwarded to the NetView program for further handling.

This chapter contains the following topics:

Events: General Information

The NetView for AIX program uses the following types of events:

Map events
: Notifications issued because a user or application does something that affects the status of the current map or of the NetView for AIX graphical interface. For example, if you add a connection between a workstation and a server on a submap, an event is generated and logged in the event log file. The contents of the submap change to include the added connection.

Network events
: A message sent by an agent to one or more managers to provide notification of an occurrence affecting a network object. These events are not necessarily reflected in the map. For example, if an SNMP agent is not in your management region, but is configured to send traps to the manager, you will receive events for that agent.

Logging Events

All of the events and SNMP traps received by the NetView for AIX program are logged in the /usr/OV/log/ovevent.log file by default. From this file, the nvevents application reads the events filtered for display and displays them through the Event History appli-

cation. See “Viewing the Event Log” on page 114 for information about using the Event History application to view events stored in this file.

SNMP traps can also be logged in the /usr/OV/log/trapd.log file. This file is in ASCII format, so you can edit it to view logged traps, or print the contents of this file. If the NetView for AIX program is configured to work with a relational database, you can transfer trapd.log data to a relational database and use the relational database tools to create reports.

Refer to the *NetView for AIX Database Guide* for more information about transferring trapd.log data to a relational database.

Information Provided by SNMP Traps

SNMP defines six generic types of traps and allows definition of enterprise-specific traps. The trap structure conveys the following information to the NetView for AIX program:

- Agent's object that was affected
- IP address of the agent that sent the trap
- Event description (either a generic trap or enterprise-specific trap, including trap number)
- Time stamp
- Optional enterprise-specific trap identification
- List of variables describing the trap

The agent knows which manager system to send traps to by use of a user-configurable trap destination. The manager system can then retrieve more information to isolate a problem by polling the agent system.

Information Provided by NetView for AIX Internal Events

The NetView for AIX program's internally generated events are treated as enterprise-specific traps. These events include the following information:

- Description of the event.
- Name of the node associated with the event in the system name (sysName) MIB variable.

  A node name with the value <*none*> refers to the manager station running the NetView for AIX program.

To look at a list of the NetView for AIX program's internally generated events, enter the **/usr/OV/bin/event -l** command from an aixterm window. See Appendix, “NetView for AIX Internal Traps” on page 241 for detailed information about the internal traps.

Starting the Event Display Application

The Event Display application can be started in any of the following ways:

- The Event Display application is automatically started when the NetView for AIX program is started. It displays all events received during the current NetView for AIX session that have been filtered for display. Only one instance of this application can be open per session to display all of that session's filtered events. However, you can open other instances to display that session's filtered events for selected network objects. You can also select events for display in a separate workspace window. Events are synchronized in client machines. That is, if an event is cleared, notes are added, or the event severity and category is changed in a workspace on a client machine, the change is reflected in the workspaces on all the clients.
- Select a symbol on a submap. Then select **Monitor..Events..Current Events** from the NetView for AIX main menu. In this case, the application displays all uncleared events (events currently displayed in the main workspace window) associated with the selected symbol.
- Click mouse button 3 on a symbol in a submap to display the object context menu and select **Monitor..Events..Current Events**. The Event Display application displays all uncleared events associated with the selected symbol.
- Select an object or objects on the submap and drag the Events icon from the Tools window to the Control Desk window or another area of the desktop. Events for each object will be displayed in separate dynamic workspaces.

You can start the NetView for AIX program without starting the Event Display application. If you are a root user, you can edit the /usr/OV/registration/C/ovsnmp/nvevents file and remove the **-Initial** flag from the command that initiates the Event Display application. If you make this change, you can still start the Event Display application from the main menu, an object context menu, or the Tools window.

When Workspaces Are Automatically Created

The NetView for AIX program automatically creates a workspace in the following circumstances:

- The first time the Event Display application is started, the workspace contains the dynamic event display for the current NetView for AIX session. It is located in the Control Desk window. You can also start this application by dragging its icon from the Tools window.
- Each time you select a node from the submap and then select **Monitor..Events..Current Events** from the main menu or drag the Events icon from the Tools window, a new workspace is opened for the node you selected.
- The first time you select **Monitor..Events..Event History** from the main menu, a new workspace that contains events from the /usr/OV/log/ovevent.log file is opened instead of events being received during the current NetView for AIX session. See “Viewing the Event Log” on page 114 for more information about the event history application.

- When you drag an event card from either the Event Display or Event History applications.
- Each time you select **Create..Dynamic Workspace** from the main workspace window. See *NetView for AIX User's Guide for Beginners* for steps on creating a dynamic workspace.

Saving the Event Workspace Configuration

You can change the following resources in the /usr/OV/app-defaults/Nvevents file so that you can save your event workspace configuration and start the Event Display application with the saved configuration:

saveEnvOnExt	Defines if the Event Display application saves all workspaces. The default value is False. When you change the value for saveEnvOnExt to True, the current event workspace configuration is saved in a configuration file in the $USER/$HOME/NvEnvironment directory when you exit the Event Display application.
loadEnvOnInit	Defines if the Event Display application starts with a saved event workspace configuration. The default value is False. When you change the value for loadEnvOnInit to True, all saved workspaces are opened inside the Control Desk when you start the Event Display application.
considerStaticWrkSpcs	Defines if the Event Display application saves and loads static workspaces in addition to dynamic workspaces. When you change the value for considerStaticWrkSpcs to True, static workspaces and workspaces that were loaded using the File..Load option are saved and opened when you start the Event Display application. The default value is False, which saves and loads only dynamic workspaces. You might want to save and load the configuration for dynamic workspaces only because starting the Event Display application with saved static workspaces can cause a delay in loading the event workspace configuration.

You can override the value set for saveEnvOnExt by using the Options..Save Environment menu bar option in the Event Display application main workspace. When the value for saveEnvOnExt is True, a radio button is displayed next to the Options..Save Environment option indicating that this option is active and the event workspace configuration will be saved when you exit the Event Display application. Select the **Options..Save Environment** option to deactivate the option, and the current event workspace configuration will not be saved. If the value for loadEnvOnInit is True, the Event Display application will start with the previously saved configuration, if one exists.

Similarly, when the value for saveEnvOnExt is False, a radio button is not displayed before the Options..Save Environment option indicating that this option is not active and

the event workspace configuration will not be saved when you exit the Event Display application. Select the **Options..Save Environment** option to activate the option and save the current event workspace configuration.

Whether the configuration is saved because the value for saveEnvOnExit is True or because you selected the Options..Save Environment option, the Event Display application is not started with the saved workspace configuration unless the value for loadEnvOnInit is True.

Controlling How Events Are Displayed

You can change the following resources in the /usr/OV/app-defaults/Nvevents file, which control how events are displayed:

workListDetailMode — Controls how events are displayed when events are in list format. The valid values are:

- 0 — Double-clicking on an event opens the card in a static workspace. This is the default.
- 1 — Double-clicking on an event opens the card in a static workspace that is resizable.

workCardDetailMode — Controls how events are displayed when events are in card format. The valid values are:

- 0 — Double-clicking on a card brings the card to the top of the stack. This is the default.
- 1 — Double-clicking on a card opens the card in a static workspace.
- 2 — Double-clicking on a card opens the card in a static workspace that is resizable.

Changing the values for the workListDetailMode and workCardDetailMode resources to 1 and 2, respectively, enables you view all the information on the card. You can resize the workspace window to resize the card in the same proportion as the window.

Suppressing Events from Unmanaged Nodes

You can suppress events from IP nodes that are unmanaged in the open map by selecting **Options..Suppress Traps from Unmanaged Nodes** from the Event Display menu. This menu option is a toggle button. To resume seeing the traps, select this menu option again.

Searching for Events

You can search for events based on predefined filtering criteria or on any or all of the following criteria:

- A search string
- An event source
- An event category
- An event severity level

Searching by Criteria

To search for events by criteria, follow these steps:

Step 1. Select **By Criteria** from the Search pull-down menu in the Event Display Application.

Step 2. Enter the search criteria in the Search String field on the Search by Criteria dialog box. You can also specify a specific event source, category, or severity to search on.

Step 3. If you want the search results displayed in their own separate workspace, select the **Create Workspace** button. Otherwise, the events are selected in the Event Display Application Main Workspace. Selected event cards are a darker shade than the others.

Step 4. Select OK to start the search and close the dialog box. Events meeting the search criteria are either selected or displayed in a Static Workspace.

Searching by Filter

To search for events by filtering criteria, follow these steps:

Step 1. Select By Filter from the Search pull-down menu in the Events Display Application.

Step 2. Select the name of the filter from the list of Available Filters in File. You can select the **File List** button to change the list of available filters. You can also edit the selected filter by selecting the **Display/Edit** button.

Step 3. Select Activate to activate the filter. The dialog box is closed. A workspace is created in the Control Desk window containing the filtered events.

Viewing the Event Log

The Event History application, which displays events from the /usr/OV/log/ovevent.log file, starts in one of the following ways:

- When you select **Monitor..Events..Event History** from the NetView for AIX main menu
- When you drag the Event History icon from the Tools window and drop it either in the Control Desk window or another location on the desktop

The displayed events come from the /usr/OV/log/ovevent.log file, which contains events from the current NetView for AIX session. The number of events that can be stored in this file at one time, and potentially displayed by the event history application, depends on the file's maximum allowable size. See “Changing the Size of the Event Log” on page 115 for more information.

When the Event History application is started, events are not displayed until you select **Query..Display Events** from the menu. This delay in displaying events gives you the opportunity to apply filter criteria to the query of the /usr/OV/log/ovevent.log file, so the display does not contain an unmanageable number of events. If there are too many

objects selected for inclusion in the filter, the filter will not be activated, and the following message will be displayed:

```
Sieve creation failed.  Error in Object creation.
```

To filter the Event History display, select **Options..Filter Control** from the menu. From the Filter Control dialog box, you can activate and deactivate filters to control the event history display. You can also use the Filter Editor to create a filter if the list of available filters does not contain one that suits your needs. See "Using the Filter Editor" on page 136 for more information.

Changing the Size of the Event Log

To change the size of the /usr/OV/log/ovevent.log file, select **Options..Set Log Size** from the Event History menu. The default log size is 128 KB. The maximum log size is 2MB. Changing the default log size through the Event History menu changes the size for the current editing session only. You can configure the ovelmd daemon to change the size permanently through SMIT. See *NetView for AIX Installation and Configuration* for those steps. The following conditions affect changing the size of the log file:

- If you specify a new size that is less than the current size of the file, the current file becomes the backup log file and a new log file is started. You might do this if you want to clear the current log file and record only events from a given time in the current session.
- If you specify a new size that is greater than the current size of the log file, events continue to be added to the current file.
- If you set the size to 0 (zero), event logging is turned off until you enter a positive integer. Events already logged will not be deleted unless the log file is deleted.

Correlating Events

You can create a ruleset that correlates or compares incoming events to event processing decisions and actions. The ruleset editor enables you to graphically create a rule comprised of event-processing decisions and actions that are represented by icons (nodes).

You can create rulesets to:

- Enhance the filtering capabilities for dynamic event displays. You can filter on MIB variables inside traps and define thresholds based on event data.
- Use object database fields to correlate events.
- Automatically remove resolved events from the Event Display application.
- Override severity and object status associated with an event.
- Automatically issue a call to a pager.

How Events Are Processed Within a Ruleset

The following list describes how the decision and action nodes process an event through a ruleset:

Node	How Event Is Processed
Decision	If the decision is determined to be true, the event is passed on to the next node (or nodes if there are multiple connections) in the ruleset. If the decision is determined to be false, processing halts on that path through the ruleset. When all processing of an event halts and the event is not passed on to the next node, the event is deleted.
Action	The action is passed to the actionsvr daemon, which starts a new process for the specified action, and the event is passed on to the next node in the ruleset. If the action does not complete successfully, the ACTF_EV (59179071) action failed trap is generated. All actions requested and the events which caused those actions are logged. Actions include: • Forwarding the event to the Event Display application • Overriding the severity or object status associated with the event • Resolving the event • Issuing any AIX command, script file or executable, or NetView for AIX command • Issuing a call to a pager • Setting a global variable to some value • Setting a NetView for AIX object database field • Setting a MIB variable

Configuring the Paging Utility

To use the NetView for AIX paging utility through an event correlation rule (using the Pager node) or from the command line (using the **nvpage** command), you need an analog line available for modem communications. Then follow these steps for the modem that is attached to your system:

Step 1. Add and configure a tty device for modem communications using the SMIT Devices..TTY option.

Step 2. Test the modem communication through the tty device using a communications program, such as ate, provided with your AIX system.

Step 3. Customize the paging utility configuration files, which are located in the /usr/OV/conf directory:

nv.carriers Lists the defined carriers. Add the appropriate entries for all paging carriers used at your site. The Numeric IDs accepted on Modem line: Y/N field indi-

cates the pager type. If numeric IDs are accepted, the pager type is numeric. If numeric IDs are not accepted, the pager type is alpha.

See the nv.carriers man page for more information.

***.modem** Contains the default information for the modem. The asterisk (*) represents the name of the modem file. The following modem files are provided:

ibm5853.modem	For the 2400 baud IBM Model 5853 modem
ibm7855.modem	For the IBM Model 7855
newhayes.modem	For most Hayes** compatible modems
oldhayes.modem	For Hayes compatible modems that do not understand the extended AT command set
qblazer.modem	For Hayes compatible modems
blank.modem	For you to copy and customize

Usually, you do not need to change the values in the modem file. If a modem file is not provided for the modem you are using, use the blank.modem file as a template.

See the modem man page for more information.

nvpager.config Lists the defaults that are the physical characteristics of the modem. Specify the tty device that you already configured and tested and change the modem characteristics to reflect the values configured on the tty device through SMIT. Add the name of the modem file that corresponds to the modem dedicated to paging. See the nvpager.config man page for more information.

After updating the configuration files, stop and restart the nvpagerd daemon to make the changes available to the paging utility.

Step 4. Create NetView for AIX security user profiles for those individuals who you want to page automatically through an event correlation ruleset. See “Creating and Changing a User Profile” on page 30 for those steps.

When you use the Pager node in an event correlation ruleset, you specify the user ID of the person you want to page. The NetView for AIX security user profile defines the user ID and the paging information. Security does not need to be activated to access the paging information in a user's profile.

You can also send a page from the command line using the **nvpage** command. See the man page for more information.

Types of Ruleset Nodes

The ruleset editor contains the following nodes as shown in Figure 20 on page 128:

Action

Specifies the action to be performed when an event is forwarded to this node. Fields from the trap being processed are available as environment variables. The specified action can be any AIX command, the full path name of any shell script or executable, or any NetView for AIX command. Usually, the output from the specified action is displayed on the screen. If the output is not displayed on the screen, it is written in the /usr/OV/log/nvaction.log file. You can use this node to execute the **/usr/OV/bin/ovxecho** command to display a dialog window when a specific event occurs.

The dialog box contains one relevant field: Action. Enter any AIX command, the full path name of any shell script or executable, or any NetView for AIX command.

Block event display

Prevents events from being forwarded to the Event Display application. Use this node if you have changed the default processing action to pass (forward) events to the Event Display application and you do not want to forward events that meet specific conditions. A trap that is processed through this node is marked so that it will not be handled by the default processing action specified for the ruleset.

The dialog box does not contain any relevant fields.

Check Route

Checks for communication between two network nodes and forwards the event based on the availability of this communication. For example, you can use this node to check the path from the Manager to a device before forwarding a node down trap.

Note: The Check Route node does not check the status of the node. It only checks the availability of the path to the node.

The dialog box contains the following relevant fields:

Source
: Specifies the node that the check route starts from.

Destination
: Specifies the node for which you are checking the route

Forward Event when
: Specifies forwarding of the event to the next node if the path is available (communication is successful) or unavailable (communication fails)

SNMP Defaults
: Specifies the following SNMP defaults: community name, number of retries, remote port, and local port

Compare MIB Variable

Compares the current value of a MIB variable against a specified value. When a trap is processed by this node, the ruleset processor issues an SNMP GET request for the specified MIB variable.

The dialog box contains the following relevant fields:

MIB Variable Name
: Specifies the fully qualified name of the variable you want to compare.

Object ID Source
: Specifies the trap attribute to be used to determine the object from which to get the specified MIB variable.

Community Name
: Specifies the community name to be used in the SNMP request.

Value to Compare
: Specifies a literal value that will be compared to the MIB variable value.

Comparison Type
: Specifies the type of comparison to be made.

You can use the **Browse MIB** button to start the MIB browser and then cut and paste information from the MIB browser into the Compare MIB Variable dialog box.

Event Attributes

Compares any attribute of the incoming event to a literal value. You can use this node to check for events generated by a particular device.

The dialog box contains the following relevant fields:

Attribute
: Specifies the name of the attribute to be compared

Comparison Type
: Specifies the type of comparison to be performed

Value
: Specifies the literal value to be compared to the specified trap attribute, such as a host name

See “Event Attribute Values” on page 127 for more information.

Forward

Forwards the event to applications that have registered to receive the output of the ruleset. A trap that is processed through this node is marked so that it will not be handled by the default processing action specified for this rule.

The dialog box does not contain any relevant fields.

Inline Action

Specifies the action to be performed when an event is forwarded to this node. Unlike a command specified in an Action node, a command specified in an Inline Action node is not sent to the actionsvr daemon. Instead, the command is executed immediately, and processing continues to the next node if the action's return code matches the return code you specify within the specified time period.

The dialog box contains the following relevant fields:

Command
: Specifies any AIX command, the full path name of any shell script or executable, or any NetView for AIX command.

Wait Interval
: Specifies the time period, in seconds, that the ruleset processor should wait for the specified action to return. Values can be from 0 to 999 seconds. If the wait interval is 0, the return code from the action is ignored and processing immediately proceeds to the next node. If a wait interval is specified, and the return code from the action is not received in the wait interval, it is considered to be a failure and processing does not proceed to the next node. If the action is not completed within the specified time period, processing will not proceed to the next node.

Command exit code comparison
: Specifies the type of comparison you want to make.

Exit Code
: Specifies the return code value from the specified action that you want to use in the comparison.

Override

Overrides the object status or severity assigned to a specific event and updates applications that have registered to receive the output of the ruleset. The Event Display application is registered to receive the output. For example, you can use this node to change the severity to Major when a node down event is received for a router. Use this node with the Query Database Field node to override status or severity for specific device types.

The dialog box contains the following relevant fields:

Status
: Specifies the new object status to be associated with this event or select **no override** if you do not want to change the status. The Event Display

application updates the object's status to this value.

Severity — Specifies the new severity level to be used for this event or select **no override** if you do not want to change the severity level.

A trap that is processed through this node is marked so that it will not be handled by the default processing action specified for this rule.

Pager

Issues a call to a pager that has been defined in a NetView for AIX user profile. You should have already configured the paging utility. See "Configuring the Paging Utility" on page 116 for those steps.

The paging utility will use the pager number and carrier information defined in the user profile.

The dialog box contains the following relevant fields:

User ID — Specifies the NetView for AIX user ID of the person to be paged. If pager information is not found in the NetView for AIX user profile or there is no NetView for AIX ID for the user, a dialog box is displayed in which you can enter the User ID and pager information. Then a user profile is created or updated.

Message Text — Specifies message text to be delivered with the page. The message can include trap data passed in environment variables. See "Environment Variables for Trap Data" on page 127 for more information.

Pass on Match

Compares some attribute of the event being processed with an attribute of all traps received in a specified period of time. You can use this node (formerly called the Correlate Event Attributes node) to check for two events that were generated by the same node in the network. The dialog box contains the following relevant fields:

Event 1 Attribute — Specifies the name of the attribute in the first event.

Comparison Type — Specifies the type of comparison to be performed.

Matching Event Attribute — Specifies the name of the attribute in the second event.

Event Retention — Specifies the length of time the first event will be held to wait for the second event. The maximum event retention value is 999 hours, 59 minutes, 59 seconds. If an attribute match is found between two events within the specified period of time, processing continues to the next node in the ruleset.

See “Event Attribute Values” on page 127 for more information.

Query Database Field Compares a value from the NetView for AIX object database to a literal value or to a value contained in the incoming event. You can use this node to check if the originating device is a router.

The dialog box contains the following relevant fields:

Field Name — Specifies the name of the database field to be queried.

Object ID Source — Specifies the source of the object ID to be used in selecting an object from the object database.

Comparison Type — Specifies the type of comparison to be performed.

Compare Field to — Specifies either a literal value or an attribute value to be used in the comparison.

Reset on Match Compares some attribute of the event being processed with an attribute of all traps received in a specified period of time. This node is similar to the Pass on Match node, except that if a match is found, the event is not passed on to the next node in the ruleset and processing stops. You can use this node to discard events before they are forwarded to the Event Display application. You might find this node useful for events that are generated from a device that frequently goes up and down. The dialog box contains the following relevant fields:

Main Event Attribute — Specifies the name of the attribute in the first event.

Comparison Type — Specifies the type of comparison to be performed.

Resetting Event Attribute — Specifies the name of the attribute in the second event.

Delay Time
: Specifies the length of time the first event will be held to wait for the second event. The maximum event retention value is 999 hours, 59 minutes, 59 seconds. If an attribute match is found between two events within the specified period of time, the event is not forwarded to the next node in the ruleset and processing stops.

See “Event Attribute Values” on page 127 for more information.

Set Database Field

Sets the value of any NetView for AIX non-boolean object database field. Fields that have TRUE or FALSE values cannot be changed.

The dialog box contains the following relevant fields:

Field Name
: Specifies the name of the field in the object database that you want to change

Object ID Source
: Specifies the source of the object ID to be used in selecting an object from the object database

Set Value to
: Specifies either a literal value or an event attribute value to be used for the database field setting

See “Event Attribute Values” on page 127 for more information.

Query Global Variable

Queries the value of the global variable that has been previously set using the Set Global Variable node.

The dialog box contains the following relevant fields:

Variable Name
: Specifies the name of the variable you are checking. The variable is created and assigned a value using the set global variable node.

Comparison Type
: Specifies the type of comparison to be performed.

Compare Variable to
: Specifies either a literal value or an event attribute value to be used in the comparison

See “Event Attribute Values” on page 127 for more information.

Set Global Variable Sets a variable for use within the ruleset itself. For example, use this node to set a flag whose value will be checked later in the ruleset using the Query Global Variable node. When the ruleset is finished processing, the global variable is no longer in effect.

The dialog box contains the following relevant fields:

Variable Name
: Specifies a user-defined text string associated with the variable's value, such as, flag

Set Variable
: Specifies one of the following settings:

- Increment Value By One

  If the global value has already been set, the value will be increased by one. If the global variable has not yet been set, the value will be set to one.

- Decrement Value By One

  If the global value has already been set, the value will be decreased by one. If the value has not yet been set, the value will be set at negative one.

- Set to Literal Value

  Enter a text value in this field.

- Set to Attribute Value

  Use the **Select** button to select a trap attribute.

See “Event Attribute Values” on page 127 for more information.

Set MIB Variable Issues an SNMP SET command to set the value of a variable in the MIB representing any network resource. For example, you can use this node to change the system contact for a particular device.

The dialog box contains the following relevant fields:

MIB Variable Name
: Specifies the name of the variable you want to change.

Variable Data Type
: Specifies the type of data to be placed in the MIB field, such as integer, string, and so on.

Object ID Source
: Specifies where to get the object ID whose MIB is to be changed

Community Name
: Specifies the community name of the object whose MIB is to be changed

Value to be Set
: Specifies a literal value to be used as the data for the MIB field value. The data type of this value must match the type specified in the Variable Data Type field.

See “Event Attribute Values” on page 127 for more information.

You can use the **Browse MIB** button to start the MIB browser and then cut and paste information from the MIB browser into the Set MIB Variable dialog box.

Resolve

Forwards a message to all registered applications indicating that a previous event has been resolved. By default, the Event Display application is registered to receive the output from rulesets. The receiving application determines how to handle a trap that has been forwarded from this node. This node is frequently used in conjunction with the Pass on Match node. You can use the Resolve node to delete an interface or node down event from the Event Display application when an interface or node up event is received. A trap that is processed through this node is marked so that it will not be handled by the default processing action specified for the ruleset.

The dialog box does not contain any relevant fields.

Set State

Sets the correlation state of an object in the NetView for AIX object database. The current state is updated in the corrstat1 field in the object database, and the previous value in the corrstat1 field is moved to the corrstat2 field. This process continues until the current state and as many as four previous states are stored in the object database. You can view the correlation state by selecting the object and then selecting the Display Correlation Status option from the context menu.

The dialog box contains the following relevant fields:

State Value
: Specifies the text string that you want to store in the `corrstate1` field of the specified object

Object ID Source
: Specifies the source of the object ID to be used in selecting the object in the object database

See “Event Attribute Values” on page 127 for more information.

Thresholds

Checks for repeated occurrences of the same trap or of traps with an attribute in common. You can use this node to forward an event after receiving the specific number of the same event received within a specific time period. Use this node with the Trap Settings node to identify a specific trap number. The dialog box contains the following relevant fields:

Type
: Specifies when the event should be forwarded by selecting one of the following values:

  First
  : When a threshold condition is reached, forwards the first *n* traps to the next node, where *n* is the number specified in the Count field

  At
  : When a threshold condition is reached, forwards the *n*th trap to the next node, where *n* is the number specified in the Count field

  After
  : When a threshold condition is reached, forwards all traps after the *n*th trap to the next node, where *n* is the number specified in the Count field

Count
: Specifies the number of traps required to reach the threshold condition

Time Period
: Specifies the length of time within which the number of events specified in the Count field must be received to reach the threshold condition. Use this field in conjunction with the Time Unit field.

Time Unit
: Specifies the unit of measure (minutes, seconds, hours, or days) for the number specified in the Time Period field.

Threshold by attribute
: Specifies the attribute in the trap to be used. By default, the threshold depends on the trap ID. See “Event Attribute Values” on page 127 for more information.

Trap Settings

Specifies a specific trap to be processed and is identified by a pair of generic and specific trap numbers.

The dialog box displays a list of enterprise names and IDs. When you select an enterprise ID, a list of generic and specific trap numbers for that enterprise is displayed in the Event Name and Specific fields. Select one or more traps from this list. The description of the trap you select is displayed in the Trap Description field. Use the **Comparison Type** button to

specify the type of comparison to be performed (equal to or not equal to).

Event Attribute Values

Use the following event attribute values to identify the event to be processed:

sysObjectID	Specifies the MIB object describing the agent's hardware, software, and so forth.
Origin	Specifies the host name generating the trap. The management station is the origin for traps generated as a result of NetView for AIX polling operations.
Generic	Specifies the generic trap value defined by SNMP.
Specific	Specifies the specific trap value defines by SNMP.
sysUpTime	Specifies the MIB system up time since the agent has been started.
Community Name	Specifies the community name.
1	Specifies the source ID and is a integer value that corresponds to the internal component of NetView for AIX that generated the event, such as netmon.
2	Specifies the host name to which this trap applies.
3	Specifies the event description and is a string value containing a description of the event that was generated.
4	Specifies specific trap data and is a string value containing internal data that is specific to the type of trap that is generated.
5	Specifies the database name and must be openview.
6-50	Varies by enterprise or trap.

You can use one word for a trap attribute. A word is a unit of text separated by blanks. To specify the word to be used, add a period (.) and the number of the word. For example, to use the second word of the traps's third variable binding, specify **3.2**. The ruleset processor will use the string that starts after the first blank and ends with the second blank.

NetView for AIX internal traps use variable bindings 1 through 5 as described in the above list. See Appendix, “NetView for AIX Internal Traps” on page 241 for more information about NetView for AIX internal traps.

Environment Variables for Trap Data

You can specify trap data using the following environment variables:

NVE	Specifies the enterprise ID
NVA	Specifies the agent address

NVG	Specifies the generic trap number
NVS	Specifies the specific trap number
NVT	Specifies the time stamp
NVC	Specifies the community name
NVATTR_<1-50>	Specifies the MIB attribute where 1-50 is the variable binding number.

These environment variables are frequently used with the Action node and the Pager node. For example, you might include a pager message similar to the following:

```
Multiple authentication failures for $NVA.
```

Sample Rulesets

The following sample rulesets are provided:

corrNdNu.rs Forwards a node down trap to nvevents and clears the event if a node up trap is received for the same device within 10 minutes.

corrIdIu.rs Forwards an interface down trap to nvevents and clears the event if an interface up trap is received for the same device within 10 minutes.

See “Activating a Ruleset” on page 131 for information about how to activate a ruleset.

Creating and Editing a Ruleset

To create a ruleset, use the ruleset editor as shown in Figure 20.

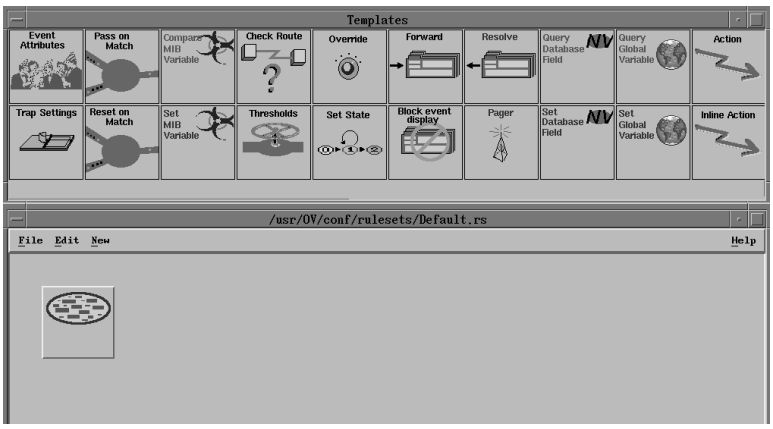


Figure 20. The Ruleset Editor

You must be a root user to start the ruleset editor. To start the ruleset editor, use one of the following methods:

- Select **Tools..Ruleset Editor** from the NetView for AIX main menu bar.

- Enter the **nvrsEdit** *ruleset_name* command at the AIX command line, where *ruleset_name* is the name of the ruleset you want to create or change.
- Double click on the Ruleset Editor icon on the tool palette.
- Drag and drop the Ruleset icon from the Tools window onto the desktop.

The ruleset editor is divided into two windows:

Ruleset	Contains the Event Stream icon, which represents all incoming events and is the work area for creating rulesets. The window's title bar contains the name of the ruleset you are currently editing. If you started the ruleset editor without specifying a ruleset name, the window's title bar contains the name `default.rs`. You can change the name of the ruleset when you save the ruleset, or you can edit an existing ruleset by selecting the File..Open option from the ruleset editor menu bar. Other menu bar operations enable you to create, modify, or delete a ruleset.
Templates	Contains the nodes that you can use to create a ruleset. Each node represents either a decision node or an action node.

Changing the Default Processing Action

You can change the default processing action, which defines what is done with a trap after if has been processed through a ruleset. The default processing action is not used for traps that have been *marked* by being processed through one of these nodes:

- Block event display
- Forward
- Override
- Resolve

If the trap is not processed through a Block event display, Forward, Override, or Resolve node, the trap can be passed on to interested applications or discarded. The default processing action is **Block**, which means that the trap is not forwarded (passed) to applications that have registered to receive the output of the ruleset, such as, the Event Display application.

To change the default processing action so that events are forwarded to registering applications, double click on the Event Stream icon and select the **Pass** button. You can override forwarding specific events that have passed through a ruleset by using the Block event display node, or you can reset the default action for all events that pass through the ruleset by selecting the **Block** button.

Adding a Node

To add a node to a ruleset, drag and drop the appropriate node from the template area onto the work area and then connect the nodes. You can also select the appropriate node from the New pull-down menu bar option. If you maximize the work area window size, select **Edit..Focus to Templates** to keep the Templates window in front of the work area window. The Focus to Templates option is also available from the context menu that is available in the background area of the work area. When two or more

decision nodes are connected sequentially (in a straight line), the logical operator AND is used. When two or more decision nodes are connected in parallel from a single decision node, the logical operator OR is used.

When you drop a node into the work area, a dialog box is displayed that contains relevant data fields for the decision or action to be performed. Complete the dialog box fields and select the **OK** button to add the node to the ruleset. The description field in each dialog box is optional. You can use this field to document the decision or action taken at the node.

You can double-click on a node at any time or select **Edit** from the context menu on a node to display its dialog box and view or modify the data.

Connecting Two Nodes

Connect the nodes in the ruleset to define the logic path through the ruleset. Connect decision nodes sequentially (in a straight line) to use the logical operator AND. Connect two or more decision nodes in parallel from a single decision node to use the logical operator OR.

Use one of the following methods to connect nodes:

- Select **Edit..Connect Two Nodes** from the ruleset editor menu bar. Then select the nodes you want to connect in the work area.
- Select the node in the work area to which you want to connect another node. Then drag and drop another node into the work area. The connection will be drawn automatically.

Select the "from" node first and then the "to" node so that the event flow through the ruleset is from left to right.

Deleting a Node

To delete a node from a ruleset, use one of the following methods:

- Select **Edit..Delete Node** from the ruleset editor menu bar and then select the node.
- Select the node and then select **Edit..Delete Selected** from the ruleset editor menu bar.
- Select the node and press the **Delete** key.
- Select **Delete** from the context menu on the node.

The node and the connection to the node is deleted.

Deleting a Connection

To delete a connection, use one of these methods:

- Select **Edit..Delete Connection** from the ruleset editor menu bar and then select the connection.

- Select the connection and then select **Edit..Delete Selected** from the ruleset editor menu bar.
- Select the connection and press the **Delete** key.
- Select **Delete** from the context menu on the connection.

Inserting Another Ruleset

You can insert another ruleset into the ruleset you are currently editing. You might find this useful for combining rulesets or building complex rules out of existing rulesets. To insert another ruleset, follow these steps:

Step 1. If you do not want the inserted ruleset connected to the Event Stream icon, select the node to which you want to connect the inserted ruleset. If you do not select a node, the ruleset is connected to the Event Stream icon.

Step 2. Select **File..Insert** from the menu bar. A dialog box is displayed containing a list of existing rulesets.

Step 3. Select the name of the ruleset you want to insert and select the **OK** button. The ruleset is inserted into the current ruleset, and the layout is recalculated.

Saving a Ruleset

When you are finished editing a ruleset, save it by selecting **File..Save** from the ruleset editor menu bar. If you want to save the ruleset with a different name than the one that is displayed in the title bar of the ruleset window, select **File..Save As** from the ruleset editor menu bar. Then enter a name for the ruleset and select the **OK** button. The name of the ruleset must include a .rs extension. You might name a ruleset that resolves node down and node up traps from the save device as `resolve.rs`. Rulesets are stored in the /usr/OV/conf/rulesets directory.

Activating a Ruleset

To activate a ruleset, create a new dynamic workspace. For each dynamic workspace, you can activate only one ruleset, and you can activate one or more filters.

Select **Create..Dynamic Workspace** from the main workspace menu bar and enter the ruleset name and any other appropriate information. The new workspace uses the ruleset and filters, if any, to determine which events are displayed. If you edit a ruleset while it is active, close and reopen the dynamic workspace window to put the changes into effect. Select the **Help** button on the Dynamic Workspace dialog box for information about the dialog box fields.

You can activate one or more rulesets for automatic action when you start NetView for AIX by editing the /usr/OV/conf/ESE.automation file, adding the names of the rulesets on separate lines. Use this method for rulesets that perform a specific action, such as a call to a pager when a particular device goes down, and you do not want to display the events.

Testing a Ruleset

When you have created and activated a new ruleset, use the NetView for AIX SMIT Diagnose..Send event to trapd daemon option to send the appropriate traps and test the results of the ruleset in the workspace you created.

A Threshold Example

Suppose you want to monitor a specific router. You can create a ruleset to display authentication failure events after a minimum of five events have been received from the router within one minute. You also want to display a message box indicating that multiple authentication failure events have been received from the router. Figure 21 shows how the completed ruleset looks.

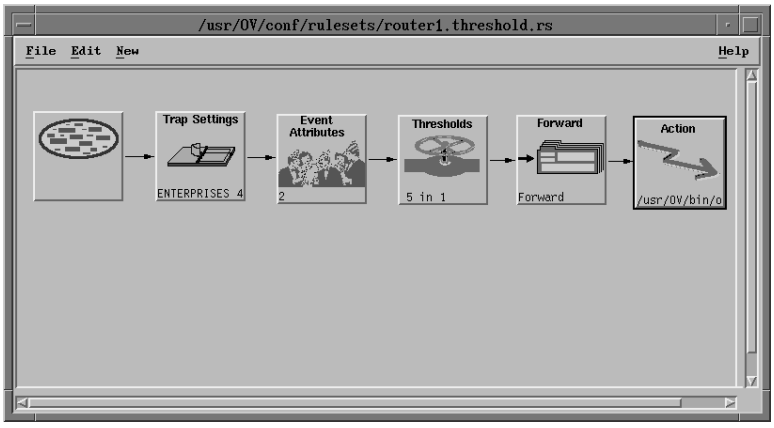


Figure 21. A Correlation Rule to Set a Threshold

Here are the steps to create and activate this ruleset:

Step 1. Select **Tools..Ruleset Editor** from the NetView for AIX menu bar. The ruleset editor is displayed.

Step 2. Drag and drop the Trap Settings node into the work area.

Step 3. Complete the dialog box as follows:

a. Select an Enterprise ID of ENTERPRISES, because this is a generic trap

b. Select Generic Trap 4 - Authentication Failure

c. You can add a description similar to the following to describe the purpose of this node:

```
Watch for authentication failure traps.
```

d. Select the **OK** button.

Step 4. Connect the Trap Settings node to the event stream node.

Step 5. Drag and drop the Event Attributes node into the work area.

Step 6. Complete the dialog box as follow:

a. Select **Origin** in the Attribute field.

b. Select **Equal To** for the comparison type.

c. Type in the fully-qualified host name of the router, for example: `route1.raleigh.ibm.com`.

d. You can add a description similar to the following to describe the purpose of this node:

```
Watch for authentication failures from router1.
```

e. Select the **OK** button.

Step 7. Drag and drop the threshold node into the work area.

Step 8. Complete the dialog box as follows:

a. Select **After** in the Type field, enter **5** in the Count field and **1** minute in the Time field.

b. Select the **Threshold by attribute** button and select **Origin** for the attribute, because you are checking for events from the same device.

c. You can add a description similar to the following to describe the purpose of this node:

```
Watch for 5 authentication failures from the
router1 within 1 minute.
```

d. Select the **OK** button.

Step 9. Connect the Threshold node to the Trap Settings node.

Step 10. Drag and drop the forward node into the work area.

Step 11. Complete the dialog box as follows:

a. You can add a description similar to the following to describe the purpose of this node:

```
Display event after five authentication failures
are received from the same device within 1 minute.
```

b. Select the **OK** button.

Step 12. Connect the forward node to the threshold node.

Step 13. Drag and drop the action node into the work area.

Step 14. Complete the dialog box as follows:

a. In the action field, enter the following:

```
/usr/OV/bin/ovxecho Multiple Authentication Failures for $NVA
```

b. You can add a description similar to the following to describe the purpose of this node:

```
Use ovxecho to display a notice when more than 5 authentication
failures are received within 1 minute.
```

c. Select the **OK** button.

Step 15. Connect the action node to the threshold node.

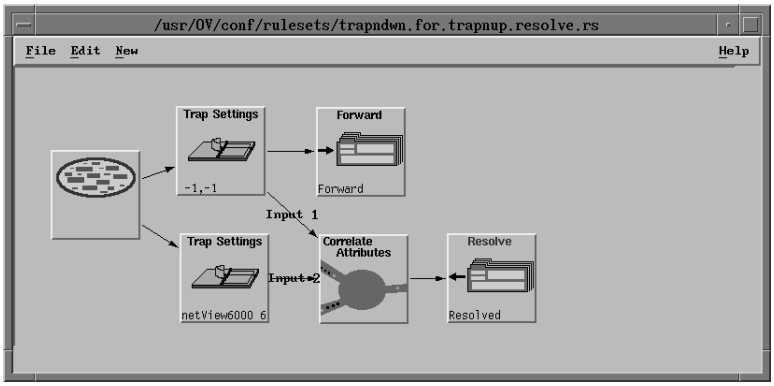
Step 16. Select **File..Save As** to save the ruleset. Enter a name for the ruleset, such as `router1.threshold.rs` and select the **OK** button.

Step 17. Activate the ruleset by creating a new dynamic workspace. Select **Create..Dynamic Workspace** from the main workspace menu bar. Enter the name of the ruleset and any other appropriate information.

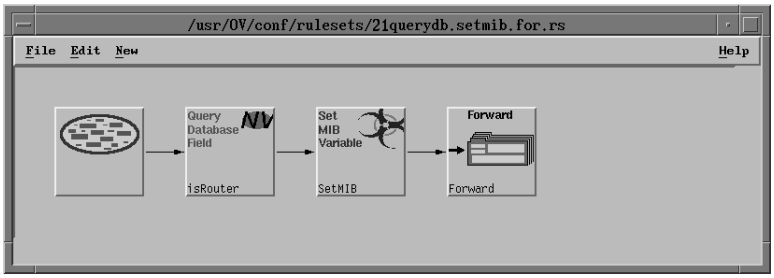
More Examples of Rulesets

Here are a few more examples of rulesets you might create. Each example includes the objective of the ruleset and a picture of how the completed ruleset would look.

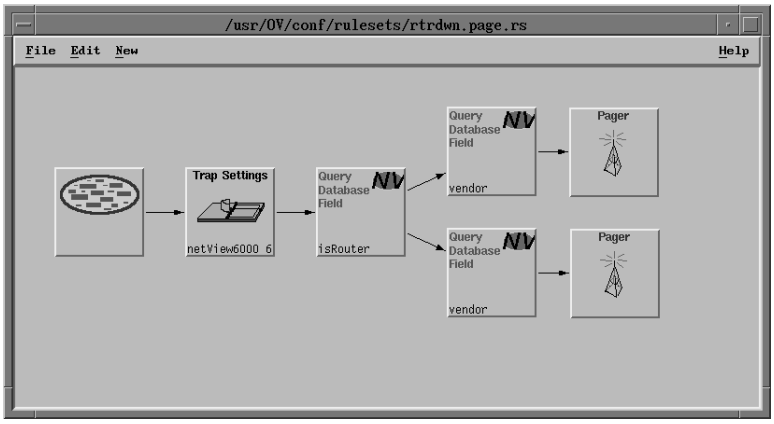
Example 1: Create a ruleset that forwards a node down trap to the Events Display application and clears the event if a node up trap is received for the same device within 30 minutes.



Example 2: Create a ruleset that changes the system contact for all routers.



Example 3: Create a ruleset that pages the appropriate individual when an IBM router or a Cisco router goes down.



This ruleset does not display the node down traps. To activate this ruleset, add the name of the ruleset to the /usr/OV/conf/ESE.automation file.

Creating Event Filters

Many events arrive at the management station on a network. Event filters are sets of criteria that determine the following conditions:

- Which events are dynamically displayed by the Event Display application.
- Which logged events are displayed by the Event History application.
- Which events are received by applications that register to receive them.
- Which events are forwarded to a host program as alerts. These filters are called trap-to-alert filters. See “Activating a Trap-to-Alert Filter” on page 145 for more information about using these filters.

Filtering events involves setting up criteria that an event must meet before it can be displayed or sent to another application. The data is then stored in the default directory, /usr/OV/filters, or in a directory that you create. To access filters, the NetView for AIX program must know the location of the directory containing the filter files.

You can create more than one filter file, and each filter file can contain one or more filters. Criteria for the same event can also be placed in several filters.

Types of Filters

The NetView for AIX program enables you to create the following types of filters:

Simple filters
: Expressions that include SNMP criteria and can be stand-alone. They can be edited using the Simple Filter Editor.

Compound filters
: Expressions that are composed of several simple filter expressions. They use nested parentheses to group simple expressions and combine them with the logical operators AND,

OR, and NOT. They can be edited using the Compound Filter Editor.

When multiple simple filters are activated, the logical OR is used. That means that traps meeting either set of filter criteria will be received. Trap exclusion is more effectively implemented by combining the two simple filters using the AND operator to create one compound filter. Then, activate the compound filter.

Accessing the Filter Editor

You can access the filter editor in any of the following ways:

- Select **Tools..Filter Editor** from the NetView for AIX main menu.
- Use the **filtered** command in an aixterm window as follows:

```
filtered -f <filename> /*name of filter file */
         -r <rulename> /*name of filter to edit*/
         -e            /*edit option; default=display*/
```

  If you want only to display a filter, you can omit the -e option. To edit a filter, use the -e option.
- Select **Options..Filter Control** in the Event Display application window. The Filter Control dialog box enables you to activate, deactivate, display, and edit filters.

Using the Filter Editor

The filter editor helps you perform the following tasks:

- Create the criteria for events you want to see.
- Organize filters in files that you specify.
- Define a threshold that enables you to modulate the number of events received over a period of time.
- Define filters that can be used independently by different applications.

When you select **Tools..Filter Editor** from the NetView for AIX main menu, the Filter Editor dialog box is displayed, as shown in Figure 22 on page 137.

Figure 22. Filter Editor Dialog Box

The name of a filter file is displayed at the top. You can work with the filters in this file or select the **File List** button to display a list of filter files from which you can choose. Once you have selected a filter file, the names and brief descriptions of all filters in the file are displayed in the bottom part of the window.

To see the contents of a particular filter, select it and select the **Display** button to the right of the list of filters. You might want to look at a filter to determine whether it is one that you want to use as a base for creating another filter or to combine it with other filters.

Select the **Add Simple** button to display the Simple Filter Editor dialog box, or select the **Add Compound** button to display the Compound Filter Editor dialog box. See “Creating Simple Filters” on page 138 and “Creating Compound Filters” on page 140 for information on these filter operations.

You might decide, after looking at a filter, that you no longer need it. In that case, select the **Delete** button to discard the filter.

To change a filter, select it from the list and select the **Modify** button. If the selected filter can be edited using the Simple Filter Editor, that editor is displayed. Otherwise, the Compound Filter Editor is displayed.

You can organize your files on a per-application basis by storing the same filter in more than one file. Use the Copy to File operation to copy a selected filter to a different file.

Note: When you create a filter, make sure that you do not include more than 250 criteria. Filters used by the Event History application are limited to 40 criteria. This maximum is the sum of all choices for the following criteria:

- Enterprises selected
- Traps selected
- Objects in the IP address list
- Selection of the event logged time criteria

- Selection of frequency parameters
- Logical operators (AND, OR, and NOT)

Creating Simple Filters

Suppose you want to create a simple filter that sends an enterprise-specific event from the NetView for AIX program to your events display. Follow these steps:

Step 1. Select **Tools..Filter Editor** from the NetView for AIX main menu.

Step 2. In the Filter Editor window, select the **Add Simple** button to display the Simple Filter Editor dialog box, as shown in Figure 23.

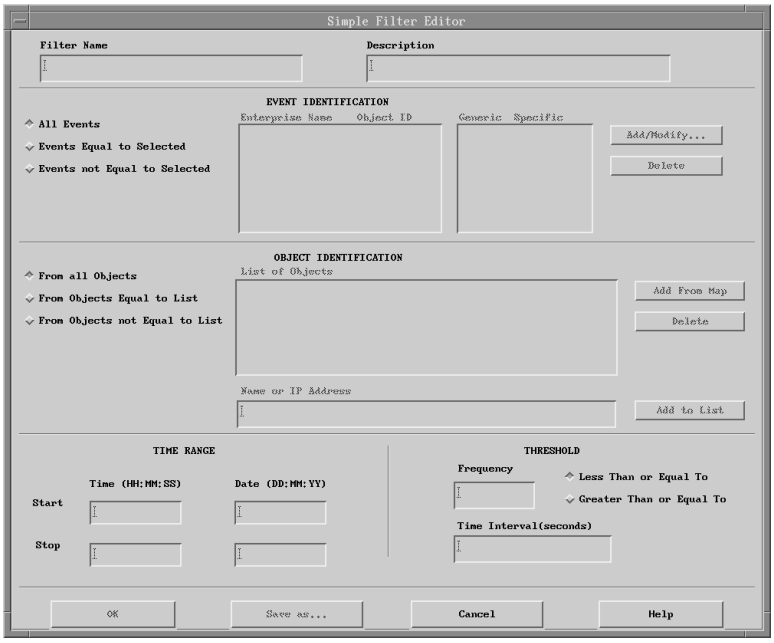


Figure 23. Simple Filter Editor Dialog Box

Step 3. Complete the Filter Name and Description fields.

Step 4. You want to receive one particular event, so select Events Equal to Selected in the Event Identification section of the Simple Filter Editor dialog box.

Step 5. Select the **Add/Modify** button to display the Enterprise Specific Trap Selection dialog box. from the list of available enterprises.

The enterprises in this list are those that are configured in the /usr/OV/conf/C/trapd.conf file. Once you make a selection from the enterprise list, all generic and specific traps associated with that enterprise are displayed in the Available Trap Types field. The default is that all traps from the selected enterprise are included as part of the filter. However, you can modify this list.

Step 6. If you know which enterprise-specific trap you want to filter, select it from the Available Trap Types list by clicking on it and then clicking on the **Select** button. The selected trap is displayed in the Selected Trap Types field. If you don't know which trap you want, go back to step 5.

If you want to add an enterprise-specific trap, enter an enterprise-specific trap number in the Specific Trap Number field and select the **Add To List** button. Select the **OK** button to apply your changes and close the dialog box.

Step 7. In the Object Identification section of the Simple Filter Editor dialog box, select From Objects Equal to List. If you selected your workstation's symbol on a submap before you opened the Filter Editor, you can select the **Add From Map** button to add your workstation to the List of Objects field. Otherwise, enter either the name or the IP address of your node to the Name or IP Address field, and select the **Add to List** button.

Step 8. You can specify time and date ranges during which this event is to be sent from your workstation to the NetView for AIX program. For example, if you want to receive this event between 8:30 and 10:30 a.m. today, complete the Time Range section as follows:

```
                TIME RANGE
        Time (HH:MM:SS)     Date (DD:MM:YY)
Start   08:30:00
Stop    10:30:00
```

The default date is today's date. If you do not specify any date or time ranges, and the filter is activated, this event is displayed every time it occurs. If you only specify a time, the date defaults to today's date, and you have to reactivate the filter each day.

Step 9. To set a threshold that specifies how many times this event may occur before notification is sent to the NetView for AIX program, complete the text fields in the Threshold section of the Simple Filter Editor dialog box.

For example, suppose you want 5 to be the maximum number of occurrences per minute of this event that can be generated without displaying the event in the nvevents application. Enter `5` in the Frequency field and select Less Than or Equal To. Enter `60` in the Time Interval (Seconds) text field.

Step 10. Select the **OK** button to close the Simple Filter Editor dialog box.

Step 11. If you want to look at the filter you just created, select the filter name from the list in the Filter Editor window and select the **Display** button.

Step 12. To activate your filter for the Event Display application, select **Options..Filter Control** from the menu in the Event Display application window. Select your filter from the list of available filters and click on the **Activate** button. Now, only those events that match your filter criteria will be displayed. When activating more than one simple filter, the logical operator OR is used. If you want the criteria of several filters to be used when filtering events, use the Compound Filter Editor to combine the simple filters.

Step 13. To select the filtered events, first select **Search..Filter** from the menu in the Event Display application window. Next, select the filter name and select the

Activate button. The search operation highlights all events that match the filter criteria. You can also create a separate workspace in which to store these events.

For more information about creating simple filters or the different sections of the Simple Filter Editor dialog box, refer to the Help system.

Creating Compound Filters

Compound filters are created by joining several existing simple filters or several simple expressions with logical operators. These filters can also be used to specify CMIS expressions that cannot be specified using the Simple Filter Editor. To create a compound filter, you need to use the Compound Filter Editor, as shown in Figure 24.

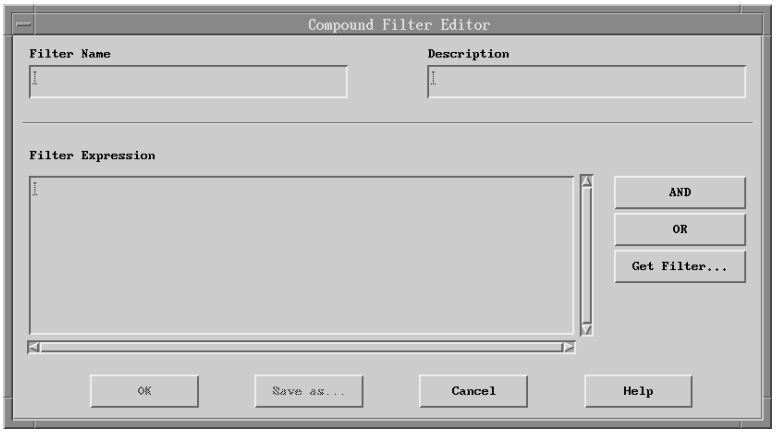


Figure 24. Compound Filter Editor Dialog Box

Follow these steps:

Step 1. Select **Add Compound** from the Filter Editor window.

Step 2. Specify a filter name and description for the compound filter.

Step 3. In the Filter Expression field, either type a filter expression or select the **Get Filter...** button.

Step 4. If you select the **Get Filter...** button, the Get Filter Dialog box is displayed. Select a filter in one of the following ways:

- To retrieve the name of a filter, select the **Get Filter Name** button.
- To retrieve the contents of a filter, select the **Get Filter Contents button**.

If you need to modify the filter in any way, retrieve its contents from the Get Filter dialog box. Otherwise, if you plan to use the filter without making changes, get only its name.

Select the **OK** button to close the Get Filter Dialog box. The selected filter or filter name is displayed in the Filter Expression field of the Compound Filter

Editor dialog box. Now you can edit this filter expression or combine it with others.

Step 5. Select one of the following logical operators to combine filter expressions:

- AND, which places && at the end of the current filter expression. Combining two filter expressions with AND means that events must match all criteria of both expressions.
- OR, which places || at the end of the current filter expression. Combining two filter expressions with OR means that events must match the criteria of at least one filter expression, but need not match the criteria of both expressions.

Step 6. Either select another filter or enter some filter criteria in the Filter Expression field. You can combine several expressions in one filter.

Step 7. Select the **OK** button to close the Compound Filter Editor dialog box.

Step 8. If you want to save your compound filter to another filter file, select the **Save as...** button and either select another file from the list or enter a new path name in the Selection field. Select the **OK** button to close the File Selection dialog box. Otherwise, select the **OK** button to close the Compound Filter Editor dialog box.

Step 9. Select the **Close** button to close the Filter Editor window.

For more information about creating and using compound filters, refer to the Help system.

Activating and Deactivating Event Filters

Filters must be activated in order to affect the destination and the display of events. Once a filter is activated, all events matching that filter pass through the active filter to the registered applications. Events that do not match the criteria are not permitted to pass through the filter. Therefore, filtering the events will reduce the number of events being displayed.

You can activate and deactivate filters for the Event Display application, the Event History application, a dynamic workspace, and the trap-to-alert conversion process. Being able to change the activation status of a filter gives you increased control over the number and type of events that are received and displayed. However, make sure that you understand the effect of activating or deactivating a particular filter before you make the change.

Filters can also be activated and deactivated programmatically. Refer to the *NetView for AIX Programmer's Guide* for information about the event filtering API.

Filtering Events for Display in the Main Workspace

You can select which events are displayed in the main workspace or the Event History workspace by activating filters.

To activate a filter, follow these steps:

Step 1. Select **Options..Filter Control** from the Event Display or Event History application's menu bar.

The Filter Control dialog box is displayed.

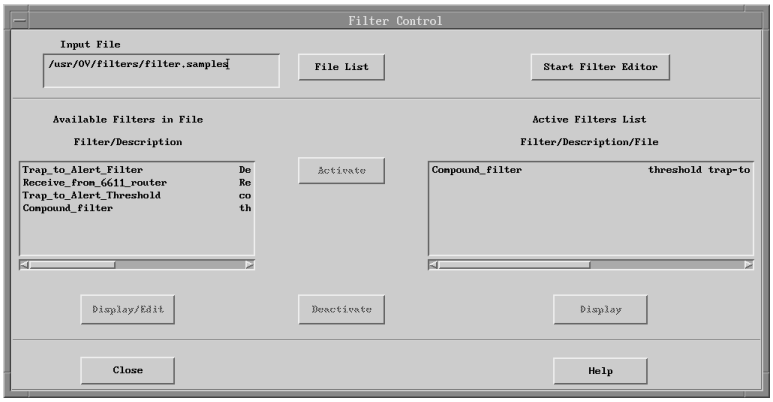


Figure 25. Filter Control Dialog Box

The Filter Control dialog box contains two sections. On the left is the **Available Filters in File** section, which lists each filter in the selected file and provides a short description. On the right is the **Active Filters List**, which displays the name, description, and full path name of all filters that are currently active.

Step 2. If you want to select a different input file, select the **File List** button and select another filter directory or file from the File Selection dialog box, shown in Figure 26 on page 143.

To see the file in the specified directory, select the **Filter** button in the File Selection dialog box. After you have made your selection, select the **OK** button to close this dialog box.

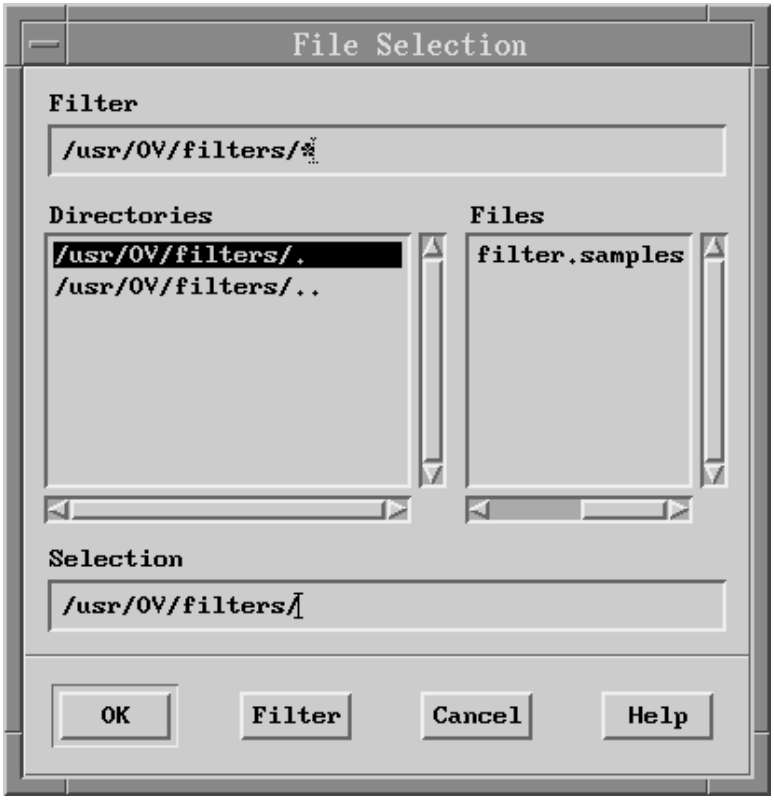


Figure 26. The File Selection Dialog Box

Step 3. To activate a filter, select a filter from the Available Filters in File Filter/Description list in the Filter Control dialog box and select the **Activate** button. The filter information is copied to the Active Filters List, and the filter is activated immediately.

Step 4. If you want display the filtered events from the /usr/OV/log/ovevent.log file for display by the Event History application, select **Query..Display Events** from the main workspace menu bar. See “Viewing the Event Log” on page 114 for more information.

If there are too many objects selected for inclusion in the filter, the filter will not be activated, and the following message will be displayed:

```
Sieve creation failed.  Error in Object creation.
```

If filters that retrieve logged events have more than 40 host names defined as one particular object, for example, IP_ADDRESS, you will receive the following error message:

```
Error receiving logged event
```

To resolve the error, edit the filter to reduce the number of objects (IP_ADDRESS).

When activating more than one simple filter, the logical operator OR is used. If you want the criteria of both filters to be used, use the Compound Filter Editor to combine them and then activate the compound filter. See “Creating Compound Filters” on page 140 for more information.

To display a selected filter from the Available Filters in File Filter/Description list, select the **Display/Edit** button below the list of available filters on the Filter Control dialog box. The Filter Editor dialog box is displayed. From this dialog box, you can make changes to the following filter criteria:

- The filter name
- The filter description
- The enterprise name
- The generic and specific event numbers
- The object from which the trap is to be sent
- The time range, including a starting date and time and a stopping date and time
- The threshold values, including frequency and time interval

Note, however, that you cannot edit a filter from the Available Filters in File list if that filter has been placed in the Active Filters List. If you need to make changes to an active filter, first deactivate the filter, then select the **Display/Edit** button to access the Filter Editor. See “Using the Filter Editor” on page 136 for more information about editing filters.

All filters you activate are saved in a user profile in your $HOME directory. The user profile for the Event Display application is called `.<userid>.events`. For the Event History application, the user profile name is `.<userid>.log`. When you start an application, the filters listed in your user profile for that application are automatically activated. When the application stops running, the profile is saved.

In addition to using the Filter Control dialog box to activate filters, you can edit your user profile outside of the application to add or remove filters.

Deactivating a Filter

To deactivate a filter, follow these steps:

Step 1. Select **Options..Filter Control** from the Event Display or Event History application's menu bar.

The Filter Control dialog box is displayed.

Step 2. Select the filter name from the Active Filters List and select the **Deactivate** button.

The filter is immediately deactivated, and the number of events being displayed might be affected.

Activating a Filter in a Dynamic Workspace

To activate a filter for a dynamic workspace, select **Create..Dynamic Workspace** from the main workspace menu bar. Refer to the *NetView for AIX User's Guide for Beginners* for more detailed information about creating a dynamic workspace.

Activating a Trap-to-Alert Filter

If you are using the NetView for AIX host connection, you can specify that selected events are to be converted to alerts and forwarded to the host NetView program. Trap-to-alert filters permit selected events to pass to the tralertd daemon, which converts events and traps to alerts. You can create and activate filters that limit the number of traps to be processed by the tralertd daemon. All trap-to-alert filters are stored in the /usr/OV/conf/tralertd_default.filter file.

Note: You must have root authority to activate trap-to-alert filters because these filters affect the configuration of the tralertd daemon.

To activate a trap-to-alert filter, follow these steps:

Step 1. Select the **Options..Event Configuration..Trap to Alert Filter Control: SNMP** operation from the NetView for AIX main menu.

The Trap-to-Alert Filter Control dialog box is displayed as shown in Figure 27.

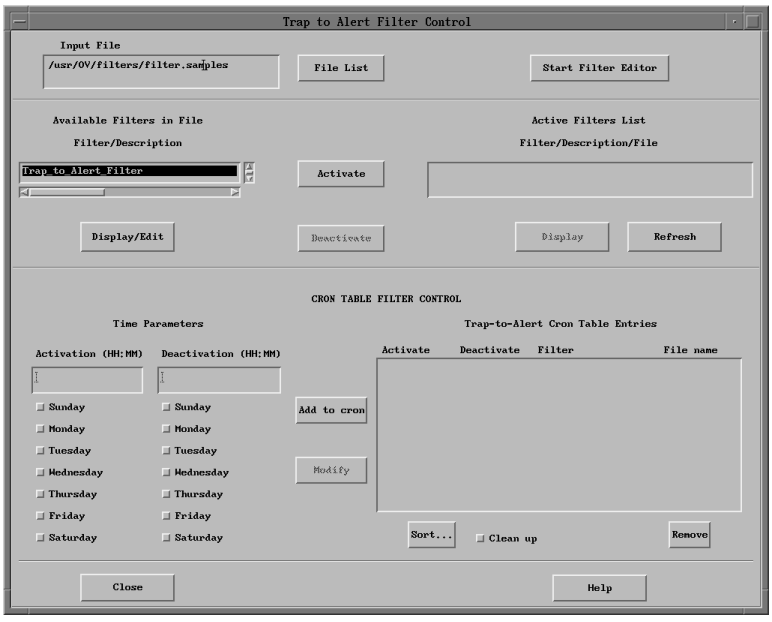


Figure 27. Trap-to-Alert Filter Control Dialog Box

Step 2. If the file name in the Input File field at the top of the Trap to Alert Filter Control dialog box is not the one you want to use, change it by using one of the following methods:

- Type a new file name in the Input File field
- Select the **File List** button and select a directory or file name from those displayed in the File Selection dialog box. Select the **OK** button to close this dialog box.

Step 3. From the Available Filters in File list, select the name of the filter you want to activate and select the **Activate** button.

The filter file name is copied to the Active Filters List and the filter is immediately activated.

If there are too many objects selected for inclusion in the filter, the filter will not be activated, and the following message will be displayed:

```
Sieve creation failed.  Error in Object creation.
```

Displaying a Trap-to-Alert Filter

Suppose you select a filter from the Available Filters in File list, but before you activate it you want to make sure that it is the one you want to use. Select the name of the filter and select the **Display/Edit** button. Depending on whether the selected filter is simple or compound, the appropriate Filter Editor dialog box is displayed. The filter criteria are displayed in the fields of this dialog box. If you need to change any fields, make the changes here, then click either on the **Save As...** button to save the changes under a different filter name or on the **OK** button to close the filter editor.

Deactivating a Trap-to-Alert Filter

To deactivate a trap-to-alert filter, select its name in the Active Filters List, then select the **Deactivate** button. The Event Display application will no longer filter events based on this filter. Deactivating a filter might affect the number of events being displayed.

Creating Cron Table Entries for Filters

Suppose you want certain filters to be activated and deactivated on a regular and frequent basis. You can accomplish this manually by using the buttons in the top section of the Trap to Alert Filter Control dialog box. However, you might spend much of your time keeping track of which filters to activate and deactivate, and when.

You can use the bottom section of this dialog box, Cron Table Filter Control, to specify automatic activation and deactivation for selected filters. This operation enables you to create and modify cron table entries. The AIX cron daemon uses these entries to control the tralertd daemon's use of selected filters. Using this operation gives you more precise control of the alerts forwarded to the host program.

Activating and Deactivating Cron Table Entries: In the Cron Table Filter Control section of the Trap to Alert Filter Control dialog box, you can enter activation and deactivation parameters for filters that appear in the Active Filters List in the top half of the dialog box. For example, to activate a filter called `EveryWednesday` on Wednesdays at 7:00 a.m. and deactivate it at 3:00 p.m. on Wednesdays, follow these steps:

Step 1. Make sure that the filter is in the Active Filters List. If it is not there, select it from the Available Filters in File list and select the **Activate** button.

Step 2. Select the filter in the Active Filters List.

Step 3. In the Cron Table Filter Control section, enter `07:00` in the Activation field and select the check button for Wednesday. Then, enter `15:00` in the Deacti-

vation field, and select the check button for Wednesday. Note that you must use a 24-hour clock for entries in these fields.

Step 4. Select the **Add to Cron** button. The activation and deactivation entries for the selected filter appear in the Trap-to-Alert Cron Table Entries field.

Step 5. Select the **Close** button to close the Trap to Alert Filter Control dialog box.

Modifying a Cron Table Entry: You can modify a cron table entry by selecting it in the Trap-to-Alert Cron Table Entries field, specifying new activation or deactivation parameters, and then selecting the **Modify** button.

Sorting Cron Table Entries: To sort the entries in the Trap-to-Alert Cron Table Entries field, select the **Sort** button. Entries can be sorted in any of the following ways:

- By activation day of the week
- By deactivation day of the week
- By activation hour
- By deactivation hour
- By filter name (the default)

To remove an entry from the list, select it in the Trap-to-Alert Cron Table Entries field, then click on the **Remove** button.

Using the selectfilter Command

You can use the **selectfilter** command in an aixterm window to accomplish the same tasks you can perform by selecting **Options..Event Configuration..Trap to Alert Filter Control: SNMP** from the NetView for AIX main menu. These tasks include the following:

- Activate a filter immediately.
- Deactivate a filter immediately.
- Add an entry to the AIX cron table that will issue the **selectfilter** command at regular intervals to activate or deactivate filters.
- Remove a selectfilter entry from the cron table.

The following example activates a filter called `newFilter` every Wednesday at 5:30 p.m.

```
selectfilter -f /usr/OV/filters
             -r newFilter
             -s /usr/OV/sockets/tralertd.socket
             -a 1 -t 17:30 3
```

For more information about the parameters of the **selectfilter** command, refer to the man page.

Configuring Events

Every enterprise has a Management Information Base (MIB) that describes operations that can be performed on that enterprise's devices in the network. Enterprises can specify traps that they expect to receive from agents that support their MIBs. You can configure the events supplied with the MIB to provide additional, more specific information about the status of network objects.

Advantages to Configuring Events

Event configuration offers the following advantages:

- You can format a trap to display information that is meaningful to you.
- You can associate an action with a trap by specifying what commands are to be executed when the management station receives an event. Configuring actions enables you to automate some fault management procedures and to restrict the amount of event information to be displayed.
- You can associate severities with an event. These severity levels are displayed in the upper-left corner of event cards. You can search on events by severity level.
- You can create new event notification categories for filtering events.
- You can create new additional actions to specify further processing that the operator should manually perform when an event is received.
- You can provide a message window to be displayed when selected events are received. This option is available when you add, copy, or modify an event. The maximum number of windows you can display is ten.
- You can access the trap-to-alert filter control to convert events to alerts that are sent to host NetView.

Customizing Traps

To use the Options..Event Configuration..Trap Customization: SNMP operation, the following conditions must be true:

- The node for which you want to configure an event must support SNMP.
- The enterprise-specific MIB for which you want to configure events must be loaded into the Loaded MIBs database.
- You must understand the definition of and purpose for the MIBs on which you want to configure events. Review the documentation provided by the MIB vendor about the enterprise-specific events included with their product.

When an event is configured, it is added to the /usr/OV/conf/C/trapd.conf file. When this event is received from an agent, the information in the /usr/OV/conf/C/trapd.conf file is used to format the trap information that is logged in the /usr/OV/log/trapd.log file.

In addition, the Event Display application reads the event information from the /usr/OV/conf/C/trapd.conf file and formats it for display in the event cards or list. The Event Display application updates symbol status based on status events. In addition to

defining the event as a status event, the object must have object or symbol status source.

By default, the Event Display application cannot update objects that have compound status source. You can determine or change the symbol's status source by selecting **Edit..Modify/Describe Symbol** from the object context menu. To enable the Event Display application to update a symbol's status if the status source is compound, change the value for the overrideCompoundStatus resource in the /usr/OV/app-defaults/Nvevents file to TRUE. When the value for the overrideCompoundStatus resource is TRUE, the Event Display application updates all symbols for the object without having to manually change the status source for each symbol. See “Indicating Symbol Status” on page 45 for more information.

To delete configured events, select the event and select the **Delete** button. Another way to delete events is to edit the trapd.conf file and delete, or comment out, selected configured events.

Steps

To configure an event that provides you with a more readable message, follow these steps:

Step 1. Select **Options..Event Configuration..Trap Customization: SNMP** from the NetView for AIX main menu.

The Event Configuration dialog box is displayed, as shown in Figure 28 on page 150. Select the **Help** button for information about the fields and buttons in this dialog box.

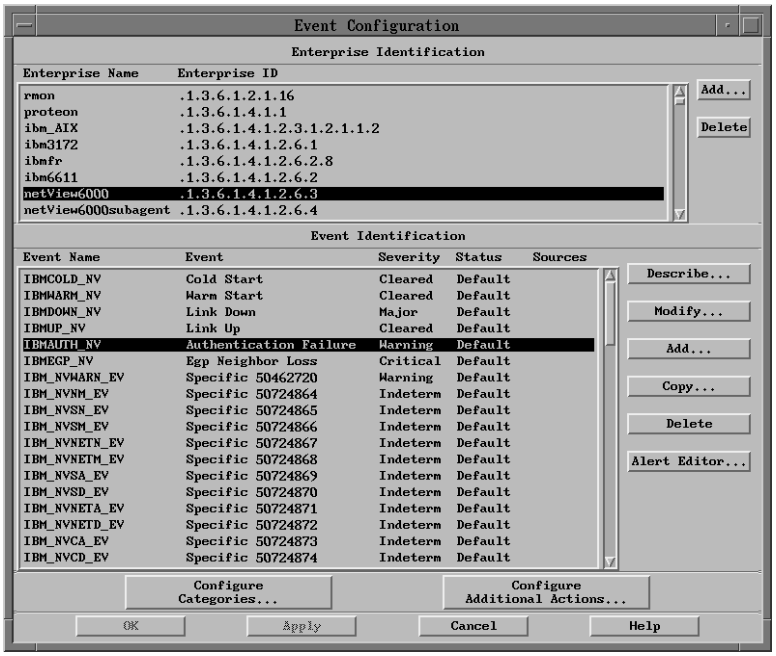


Figure 28. Event Configuration Dialog Box

Step 2. If desired, use the Configure Event Categories dialog box to define event categories. To display this dialog box, select the **Configure Categories** button.

Step 3. In the Event Identification section, select an enterprise by selecting an item in the selection list. If the enterprise whose event you want to configure is not in this list, select **Add New Enterprise...** and enter the enterprise name and object ID in the Add New Enterprise dialog box. Select the **Add** button to add the new enterprise and close the dialog box.

To view a list of the NetView for AIX program's internally generated events, enter the **/usr/OV/bin/event -l** command from an aixterm window.

Step 4. Select the event that you want to configure from the selection list in the Event Identification section. If the event already exists, select the **Modify** button to display the Modify Event dialog box. Otherwise, you can create a new event by selecting the **Add** button or the **Copy** button. Selecting the **Add** button displays the Add Event dialog box. Selecting the **Copy** button displays the Copy Event dialog box.

Step 5. Complete the fields in the Add Event, Copy Event, or Modify Event dialog box. Select the **Help** button in each dialog box for detailed information on completing the fields. Select the **OK** button to return to the Event Configuration dialog box.

Step 6. Repeat steps 3 through 5 until all events have been configured.

Step 7. Use the Configure Additional Actions for Operator dialog box to define additional actions that the operator should manually perform when an event is

received. Select the **Configure Additional Actions** button to display the dialog box.

Step 8. Select the **OK** button or the **Apply** button in the Event Configuration dialog box to apply any changes you have made. Select the **Cancel** button if you want to cancel the changes.

Note: Behaviors such as list sizes and default values can be modified in the /usr/OV/app-defaults/XNm file.

Verifying Trap Customization

If you want to verify the change, look at the /usr/OV/conf/C/trapd.conf file. The next time this event is received from an agent, the log message will contain the message you specified.

If you want to make sure that the message is displayed as you specified, you can issue the **snmptrap** command in an aixterm window. The **snmptrap** command issues an SNMP trap based on the parameters you specify in the command. For example, to send the NetView for AIX enterprise-specific trap number 59160427 to a host named `host1` from an agent named `agent1`, enter the following command:

```
snmptrap host1 "" agent1 6 59160427 ""
```

For more information about the **snmptrap** command, refer to the man page.

Another way to see the event you just created is to use the **event** command. This command sends an event to the trapd daemon. Specify the specific number of the event as follows:

```
event -E 59160428
```

For more information about the **event** command, refer to the man page.

Using the addtrap Command to Configure Events

You can type the **addtrap** command in an aixterm window to configure an event without using the Options..Event Configuration..Trap Customization: SNMP menu item. The **addtrap** command creates a trap and adds the new trap to the /usr/OV/conf/C/trapd.conf file.

If there is no enterprise definition for the trap, the new enterprise definition is added. If a trap exists with identical enterprise-object-ID, generic-trap, and specific-trap values, the **addtrap** command updates the existing trap with the new information.

Example

The following example illustrates the **addtrap** command that adds a trap for the IBM 6611 Router to the /usr/OV/conf/C/trapd.conf file.

```
addtrap -n ibm6611
        -l mytrap
        -i .1.3.6.1.4.1.2.6.2
        -g 6 -s 16 -o A -t 3
        -c "Threshold Events"
        -f !
        -F '$E $G $S $T'
        -S 3
        -C xecho
        -A 'Threshold Event received from 6611 agent $E $G $S'
```

This command specifies the following information:

-n The enterprise name is `ibm6611`.

-i The enterprise ID is 1.3.6.1.4.1.2.6.2

-g The generic trap number is 6.

-s The specific trap number is 16.

-o The trap is sent from an agent, in this case, the 6611 router agent.

-t The object that generates the trap is to be assigned a status of Marginal on the map.

-c This is a threshold event.

-f A specified action (see -C and -A below) will be performed by the management system when this trap is received.

-F The enterprise name ($E), generic ($G) and specific ($S) event numbers, and the time-stamp ($T) are displayed in the event cards or list.

-S The trap is a Severity 3 (Minor) trap.

-C The **xecho** command is activated when this event is received.

-A The following arguments are passed to the NetView for AIX program with this event:

- Event text (`'Threshold Event received from 6611 agent'`)
- The enterprise name ($E)
- The generic trap number ($G)
- The specific trap number ($S)

For more information about the **addtrap** command, refer to the man page.

Displaying a Warning Window for Events

To display a pop-up window, specify a shell script when a specific event occurs that executes the **ovxbeep** or **ovxecho** command when a specific event occurs. If you execute the ovxbeep command in the shell script, an error dialog box is displayed with an audible alarm. If you execute the ovxecho command in the shell script, an error dialog box is displayed without an audible alarm. The shell script must export the display to the appropriate workstations before executing the ovxecho or ovxbeep com-

mands, and the xhost command must have been run on the workstations where the pop-up window is to be displayed.

You specify the name of the shell script in the Optional Command and Argument format section of the Event Configuration dialog box.

For example, let's say you want to display a pop-up window when NodeA or NodeB fail. For NodeA you want to include an alarm. You also want to send an electronic-mail notice of the failure. Here are the steps:

Step 1. Select **Options..Event Configuration..Trap Customization: SNMP** from the NetView for AIX main menu.

Step 2. On the Event Configuration dialog box, select or enter the following:

Enterprise name: netview6000

Event: Specific 58916865

Step 3. Select **Modify**. The Modify Event dialog box is displayed.

Step 4. Enter the following in the Command for Automatic Action field:

```
< ShellScriptPath > $2
```

Step 5. Select **OK** to close the Modify Event dialog box.

Step 6. Select **OK** to close the Event Configuration dialog box.

Note: If you filter out an event for which you have configured a command for automatic action, the actions specified in the shell script will still be executed. If the shell script executes the ovxbeep or ovxecho command, for example, an error dialog box is displayed even though the event has been filtered out.

Example Shell Script

Following is the shell script used for the above example.

```
#!/bin/ksh
# example.sh
#
# Shell script for node down trap from the netview6000 enterprise
# (specific = 58916865). Displays warning messages and sends e-mail.

export DISPLAY=NodeA.raleigh.ibm.com:0
export DISPLAY=NodeB.raleigh.ibm.com:0

if [ $1 = NodeA.raleigh.ibm.com ]; then
  /usr/OV/bin/ovxbeep $1" is down"
  echo $1" is down" | mail oper1@manager.raleigh.ibm.com
fi
if [ $1 = NodeB.raleigh.ibm.com ];then
  /usr/OV/bin/ovxecho $1" is down"
  echo $1" is down" | mail oper2@manager.raleigh.ibm.com
fi
```

The $2 passes to the script the name of the device that generated the alert. The shell script checks the $2 flag to see whether it is NodeA or NodeB that generated the alert. If it is NodeA, the shell script calls a program,/usr/OV/bin/ovxbeep that displays a window and an audible alarm. If it is NodeB, the shell script calls the program, /usr/OV/bin/ovxecho that displays a window without a sound. For either node, an electronic-mail notice is sent to the addresses specified in the shell script.

See the /usr/OV/prg_samples/nnm_examples/beeper/beep_951x sample shell script for more examples.

Converting Events to Alerts

If you are using the NetView for AIX host connection, you can edit the events, or traps, that are converted to SNA alerts and forwarded to the host program.

Use the Alert Editor to define the SNA alert for a trap that must be forwarded to the host program. You can start the Alert Editor from the Options..Event Configuration..Trap Customization: SNMP menu item, which displays the Event Configuration dialog box. The Alert Editor button is on the right side of this dialog box.

Use the Alert Editor to perform the following tasks:

- Configure the trap-to-alert mapping for selected events.
- Define the alert for each expected trap by specifying the following information:
  - Type of event
  - Description
  - Probable or failure causes
  - Qualifiers
  - Recommended actions to take
- Delete an alert.
- Write alerts to the tralertd.conf file, which can be edited.
- Check errors to make sure that the defined alert is valid.

The Alert Editor also enables you to modify existing information on alerts through text fields and buttons. When the events are converted and sent to the host program, pertinent alert information is displayed on host NetView screens.

Using the addalert command

You can use the **addalert** command to add an alert definition to the tralertd.conf file without using the NetView for AIX graphical interface. Enter the **addalert** command and its parameters in an aixterm window to perform the same function as selecting Options..Event Configuration and selecting the **Alert Editor** button.

The following example of the **addalert** command adds an alert definition for a SynOptics** agent.

```
addalert -o .1.3.6.1.4.1.10
         -l synopalert
         -g 6
         -s 0
         -t 1
         -d 1400
         -p 0202
         -q 13
         -m "$1"
         -f 0202
         -a 1320
```

This command specifies the following information:

-o The enterprise ID is .1.3.6.1.4.1. 10

-l The alert label is synopalert.

-g The generic trap number is 6.

-s The specific trap number is 0.

-t The alert type is Permanent (1).

-d The Generic Alert subvector (X'92') contains a description for generic alert 1400, Loss of Electrical Power.

-p The Probable Causes Alert subvector (X'93') contains probable cause 0202, Internal Power Control Unit.

-q The Detailed Data Alert subvector (X'98') contains the detailed data entry Status Code.

-m The message passed to the host program resides in $1.

-f The Failure Caused Alert subvector (X'96') contains failure cause 0202, Internal Power Control Unit.

-a The Recommended Actions subvector (X'81') contains recommended action 1320, Check Cable Connection and Retry.

For more information about the **addalert** command, refer to the man page.

Refer to *NetView for AIX and the Host Connection* for instructions on using the Alert Editor.

Sending Alerts to the Host Program

When a trap is received by the trap-to-alert conversion process and it matches an active filter, the trap is converted to an SNA alert and forwarded to the NetView program.

The goal is to send all information about a trap to the host program. However, sometimes the trap contains too much information to be sent in one piece. In that case, the trap is saved in the tralertd database and assigned a Log ID, which is sent to the host

program in the alert. NetView uses the Log ID to issue the **gettrap** command within a RUNCMD to request complete trap information for the incomplete alert.

The trap information sent to the host program differs based on whether the trap is IBM enterprise-specific, non-IBM enterprise-specific, or generic. Refer to *NetView for AIX and the Host Connection* for more information about sending alerts to NetView.

Chapter 6. Managing Network Configuration

One of the challenges of network management is keeping track of all the devices on a network and ensuring that you have current information about how they are configured. Current configuration information can help you perform the following tasks:

- Make sure that all devices are configured correctly.
- Resolve network connectivity, performance, and service problems.
- Customize polling intervals to regulate network traffic and collect necessary information.
- Configure SNMP proxies to manage non-SNMP devices.

This chapter describes the following configuration management tasks:

Discovering the Network

The NetView for AIX program provides two ways to discover IP networks and enables the discovery of open topology networks. Network discovery provides you with a database of network configuration information. This section describes the following:

Automatic Network Discovery

The NetView for AIX program uses an automatic network discovery process to generate and maintain a network topology database. The more nodes on the network that support an SNMP agent, the more efficient this discovery process will be, and the more complete and accurate the resulting configuration information will be. Discovery starts with the management station, then proceeds to discover everything up to the first set of routers. Subnets beyond that are unmanaged. Use a seed file to include additional devices in the initial discovery process.

Information Retrieved

When a new node is discovered, it is added to the topology database and also to the list of nodes that is being monitored. If the newly discovered node supports an SNMP agent, information about its system configuration is retrieved and stored in the database. Table 13 on page 158 shows the information that is retrieved:

Table 13. Configuration Information Retrieved From Nodes During Discovery

Information	MIB Variable	Description
System description	sysDescr	Includes the full name and version of the system's hardware type, software operating system, and networking software.
System object ID	sysObjectID	Identifies the network object's place in the MIB hierarchy.
Forwarding status	ipForwarding	Indicates whether this entity is acting as an IP gateway to forward datagrams received by, but not addressed to, this entity.
IP address table	ipAddrTable	Lists addressing information relevant to this entity's IP addresses.
Interface table	ifTable, ifNumber	Lists interface entries by number.
System location	sysLocation	Indicates the physical location of this network object.
System contact	sysContact	Lists the contact person for this network object and tells how to contact that person.

Once the node has been discovered, these MIB values are polled periodically. Any changes are reflected in the topology database.

If the NetView for AIX program is configured to work with a relational database, you can store IP topology data in a relational database and use the relational database tools to create reports. Refer to the *NetView for AIX Database Guide* for more information about transferring IP topology data to a relational database.

You can store additional, enterprise-specific information with each node and network. This information, used with the predefined data generated by the NetView for AIX program, can give you a clearer picture of your network's configuration.

Note: NetView for AIX sometimes thinks that devices with multiple interfaces are routers and displays the router symbol.

Turning Off Automatic Discovery

To turn automatic discovery off, select **Topology/Status Polling Intervals: IP...** from the **Options** pull-down menu. Select the **New Node Discovery Switch** button to turn polling off.

Using a Seed File to Control Network Discovery

When NetView for AIX is started for the first time, the default management region is the management station on which the NetView for AIX program is operating and any networks to which it is attached. The discovery process generates the topology map by working outward from the management station up to the first set of routers.

You can define a management region by using a *seed file*. A seed file contains a list of host names or IP addresses of SNMP nodes within your administrative domain. Using a seed file forces or restricts the discovery process to generate the topology map beginning from nodes other than the management station. See *NetView for AIX Installation and Configuration* for information about configuring a seed file.

Discovering Open Topology Networks

The NetView for AIX program uses specially created applications to manage networks that use protocols other than IP. These applications pass topology information to the NetView for AIX program in the form of enterprise-specific SNMP traps or API calls.

An open-topology discovery application can be started whenever the NetView for AIX program is started, or it can be started only when a node that supports a protocol other than IP is discovered on the network. The information acquired by the open-topology discovery application is stored in the general topology database on the manager, and can be displayed on a submap along with information about IP network nodes.

Discovering Topology Using the Openmon Application

You can use the NetView for AIX openmon function to discover and load non-IP topology information from an openmon agent into the NetView for AIX program. Using the Administer..Openmon Application operation, you can start the openmon configuration program to create an application that interacts with an agent that represents a specific non-IP topology. The application stores the information obtained from the agent in the general topology database. The topology is displayed along with other network topologies, such as IP, on the NetView for AIX graphical interface. The topology information is correlated with IP topology information to determine whether an IP object can also be identified as having an association with a non-IP protocol. Using the Administer..Openmon Application operation, you can also start, query, or stop the application.

Refer to the openmon man page for additional information.

The Novell NetWare** topology is an example of a topology that openmon creates. The openmon function interfaces with one or more Novell NMS/Export Services to discover and load the Novell Netware topology into the general topology database.

The NetWare topology is represented on the root map with a Novell NMS icon. When you double-click on the Novell NMS icon, an NMS icon is displayed for each NMS/Export Service. Each NMS/Export Service contains one object for the IPX** network view and one object for the servers view. Here is a description of the NetWare topology views:

IPX network	Contains all the IPX network segments connected by the routers that NMS discovered.
Servers	Contains the icons for all the NetWare servers that NMS discovered
Router	Contains the icons for the router adapters and the running software functions, such as IP, IPX routing functions, file servers, and so forth.
Segment	Contains all the NetWare servers, requesters, hubs, and routers in the segment
Server	Contains the icons for the adapter and all the running functions such as file server, printer server, and so forth.
Requester	Contains the icon for the adapter
Hub	Contains the icons for the adapter and the running functions such as the hub function, file server, and so forth.

Configuring Symbol Creation Time and Buffer

The ipmap application draws the IP topology maps that represent your network in the graphical interface. When a node is discovered, ipmap stores it in a buffer. When the buffer reaches its threshold, ipmap draws the symbols for the nodes in the buffer onto the map. In other words, it dumps the buffer to the map. Depending on the size of your network and the speed at which nodes are discovered, you can improve the performance of ipmap by altering the size of the buffer and how often it is dumped.

Using the **File..Describe Map** option, you can configure the number of symbols to be created at one time and how often they are to be created. When a node is discovered, it remains in the buffer until the maximum number of nodes arrives or the set time has expired. Normally, the batch size is never reached before the time expires.

The default for the buffer size is 150 nodes. The maximum value allowed is 1000 nodes and the minimum is 1. The synchronization buffer uses the same value set for the buffer size (when no timer is present).

The default and recommended setting for the timer is 3 seconds. More than 3 seconds results in poor response time and less time results in slower overall performance of ipmap. The maximum value allowed is 3600 seconds and the minimum is zero (0).

The size of the buffer should vary in proportion to the speed of the processor. The following buffer sizes are recommended:

Processor	**Buffer size limit**
System/320	100
System/340	150
System/350	250
System/370	400

Steps

To change the buffer size or timer, follow these steps:

Step 1. Select **Describe Map** from the File pull-down menu.

Step 2. Select IP Map from the Configurable Applications selection list.

Step 3. Select **Configure For This Map**.

Step 4. Enter the number of symbols in the field **How many symbols should be created at one time?**.

Step 5. Enter the time in the field **How Often (in seconds) should IP Map create symbols?**.

Step 6. Select **Verify** to check what you entered.

Step 7. Select **OK** to apply the change and close the dialog box.

Increasing the ovwdb Cache Size

The ovwdb daemon acts as a caching daemon for the object information stored in the object database. You can control the number of objects maintained in the cache. Increasing the cache size will improve your CPU performance, particularly for networks containing more than 5000 objects.

Changing the Size

Change the size of the ovwdb's cache from the default of 5000 objects to a number larger than the number of objects in the ovwdb database. To get the number of objects in the ovwdb database, use the following command:

```
ovobjprint | head -1
```

To change the size of the ovwdb's cache, follow these steps:

Step 1. Exit all NetView for AIX windows. All opened sessions of the ovw application are closed during execution of the last step.

Step 2. Access NetView for AIX SMIT.

Step 3. Select **Configure**.

Step 4. Select **Set options for NetView for AIX daemons**.

Step 5. Select **Set options for topology, discovery, and database daemons**.

Step 6. Select **Set options for ovwdb daemon**.

Step 7. Enter a number in the Number of objects to hold in cache field. Enter a number larger than the number of objects defined in the ovwdb database. That allows the cache to grow to the maximum size if needed.

Step 8. Select **Do**. Any NetView for AIX applications that are running will be stopped.

Monitor the size of the database and adjust the cache size as necessary. If RAM size and paging space are not a problem, using a cache size of zero allows the cache size to grow to an unlimited size and shrink as needed.

If NetView for AIX discovers a network that exceeds the available paging space, AIX may stop a process, including ovwdb, to relieve paging space. See “Monitoring File System and Paging Space” on page 201 for information about monitoring your paging space.

Monitoring Network Configuration

There are many reasons to keep track of the configuration of objects on your network. The following list provides some configuration information you might need:

- What the IP and non-IP addresses are for a node or nodes
- What types of interfaces a node supports, and the status of each
- How a node is connected to the network
- Whether two nodes show the same address for a third node
- Whether a particular service has been installed on a remote node

The following sections describe how you can obtain this information.

Listing IP Addresses for Remote SNMP Nodes

To list IP and non-IP addresses associated with a remote SNMP node, select a node on a submap, then select the **Monitor..Network Configuration..Addresses** option from the NetView for AIX main menu. This operation collects information that you might otherwise have to obtain by looking at numerous configuration files. The following information is provided about each interface this object has with the network:

- The index, or MIB instance, of the interface on the selected node. You must have this value if you plan to set up MIB data collection.
- The name of the interface
- The IP address of the interface
- The network mask
- The network address
- The link address, if any

Checking Configured Interfaces

Use the **Monitor..Network Configuration..Interfaces** operation to list information about interfaces on remote SNMP nodes. This information can help you resolve performance problems because it provides statistics on incoming and outgoing SNMP node traffic and associated errors. It can also help you resolve connectivity problems, because it lists the status of interfaces.

The Monitor..Network Configuration..Interfaces operation provides the following information:

- The index, or MIB instance, of the interface on the selected node. You must have this value if you plan to set up MIB data collection.
- The name of the interface.

- The type of interface, for example, loopback, Ethernet, FDDI.
- The maximum transmission unit (MTU) size. This is the largest packet size that can be sent unfragmented.
- The status of the interface, which can be `up`, `down`, or `testing` (no operation packets can be passed through the interface).
- The total number of input packets and the number of erroneous input packets received.
- The total number of output packets and the number of erroneous output packets sent.

For hubs and bridges, each entry in the table corresponds to a port on the hub or bridge.

Viewing Routing Table Information

To obtain routing table information for selected remote SNMP nodes, use the **Monitor..Network Configuration..Routing Table** operation. This operation can help you resolve connectivity problems. The following information is provided by this operation:

- Destinations. The default destination is a route used by the system when it cannot find a specific route.
- The name of the next gateway between the selected node and the destination.
- The type of connection, as follows:
  - Direct—To a directly connected local area network (LAN)
  - Remote—Through a remote gateway
  - Other
- The name of the interface that is used to reach the destination.

You can also list information about gateway routing tables by entering the following command in an aixterm window:

```
rnetstat -r <host name>
```

Refer to the man page for more information about the **rnetstat** command.

Obtaining ARP Cache Information

The Address Resolution Protocol (ARP) cache is helpful in resolving connectivity problems, because it can tell you whether two nodes have a different link address than a third node. To obtain this information for selected remote SNMP nodes, select **Monitor..Network Configuration..ARP Cache** from the NetView for AIX main menu. The following information is displayed:

- The name or IP address of the destination node
- The link address associated with the destination node

- The interface name of the selected node that is used to access the destination node

You can list the ARP Cache table for a selected SNMP node by entering the following command in an aixterm window:

```
rnetstat -A <host name>
```

For more information about the **rnetstat** command, refer to the man page.

Listing Configured Services

You might need to find out what network services a node is configured to support. For example, a user might be having trouble accessing a particular service on a remote SNMP node. Use the **Monitor..Network Configuration..Services** operation to learn the following information:

- The service protocol, either TCP or UDP
- The port to which the service is bound
- The service for which the node is listening, such as SNMP, telnet, or NFS. If this field is blank, the service is unknown.

You can get a listing of configured services for a remote SNMP node by entering the following command in an aixterm window:

```
rnetstat -S <host name>
```

For more information about the **rnetstat** command, refer to the man page.

Retrieving MIB Configuration Information

Certain MIB variables store information that provides a summary of the configuration of a selected network node or nodes. The operations described in this section enable you to conveniently retrieve the values of frequently accessed MIB variables.

Displaying MIB Interface Information

To display the current values of a network object's interface MIB variables, select a symbol on the submap, then select **Monitor..MIB Values..Interface Info...** from the NetView for AIX main menu. The following information is displayed:

MIB Variable	Description
ifDescr	Information about the manufacturer, product, and version of the hardware interface
ifType	Type of interface
ifMtu	Size of largest datagram that can be sent or received over this interface, specified in octets
ifSpeed	Estimated current bandwith of the interface, in bits per second (bps)

ifPhysAddress	Address of protocol layer at the protocol layer immediately below the network layer in the protocol stack, in octets.
ifAdminStatus	Desired state of the interface (*up, down, or testing*)
ifOperStatus	Current operational state of the interface

Displaying MIB System Information

To display the current values of a network object's system MIB variables, select a symbol on the submap, then select **Monitor..MIB Values..System Information...** from the NetView for AIX main menu. The following information is displayed:

MIB Variable	Description
sysDescr	Information about the name and version of the hardware, software operating system, and networking software
sysObjectID	The node that is being managed
sysUpTime	Time, in hundredths of a second, since the network management portion of a system was last initialized
sysContact	Contact person for this managed node, and how to contact that person
sysName	Administrative name (fully qualified domain name) assigned to this node
sysLocation	Physical location of the node

Setting and Changing Polling Intervals

You can turn polling on or off and set polling intervals for all IP nodes in the management region by using **Options..Topology/Status Polling Intervals: IP**. To modify polling intervals, you must have either root authority or write permission for the configuration file /usr/OV/databases/openview/topo/polling. Polling intervals apply to all nodes; you cannot use this option to set polling intervals for individual nodes. If you want to set or change polling intervals for individual nodes, select either **Tools..Data Collection & Thresholds: SNMP** or **Options..SNMP Configuration**. The Enable Polling and Discovery Settings button must be selected to change polling interval values.

Figure 29 on page 166 shows the Topology/Status Polling Configuration Dialog Box.

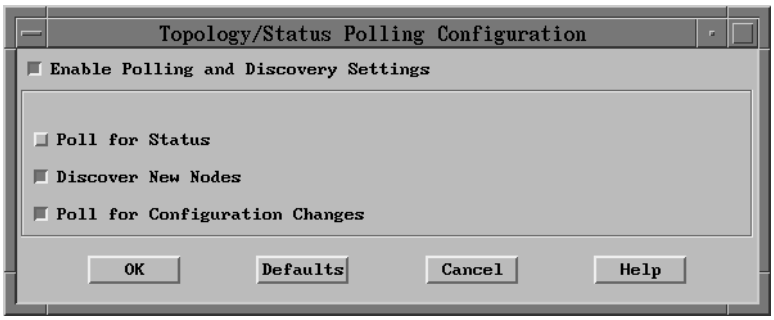


Figure 29. The Topology/Status Polling Configuration Dialog Box

To turn off polling intervals and discovery, regardless of how they are set individually, deselect the Enable Polling and Discovery Settings button.

Note: The polling configuration values are set to the default settings when automatic map generation is restarted or the databases are cleared.

To change the values in the other fields of the Topology/Status Polling Configuration dialog box, the Enable Polling and Discovery Settings button must be selected. The values you set here apply to all nodes except the ones you have individually set.

Turning Polling On and Off

To turn status polling on or off, select the **Poll for Status** button. If you turn status polling on, nodes are polled for status states. If they do not respond to status polling during the amount of time specified in the Delete Nodes Down for field, they are automatically deleted from the object database. The default is 7 days. To change this field, the **Poll for Status** button must be selected.

You can turn polling for new nodes on or off by selecting the **Discover New Nodes** button. When this button is selected, NetView for AIX polls existing SNMP nodes to determine if new nodes exist. The frequency of this polling is determined by whether you use an auto-adjusting polling interval or specify a fixed polling interval. Use an auto-adjusting polling interval to send less polling traffic to the manager once most of the network is discovered. The Use Auto-Adjusting Polling Interval button must be deselected to enter a fixed polling interval value.

Performing Configuration Checks

SNMP nodes can be polled at specified time intervals to check their configuration status. You can regulate this polling by selecting the **Poll for Configuration Changes** button and specifying a value in the Polling Interval field.

By default, the NetView for AIX program performs configuration checks once per day. It checks the /usr/OV/conf/ovsnmp.conf file to determine how frequently to poll the node for status. This file contains default configuration values, which you can change; see “Configuring SNMP Nodes” on page 167 for more information.

Configuration checking provides the following information for selected network nodes:

- Change in contact or location
- Forwarding IP packets change
- Interface added
- Interface deleted
- Incorrect routing by a node
- Link address change
- Mismatch of link address
- Network mask change
- Node name change
- Object identifier change
- Undetermined link address

Configuring SNMP Nodes

The NetView for AIX program's default configuration values are stored in the /usr/OV/conf/ovsnmp.conf_db file. You can make the following changes to this file by selecting the **Options..SNMP Configuration** operation from the NetView for AIX main menu:

- Change the default SNMP configuration.
- Change the netmon daemon's status polling intervals.
- Configure SNMP proxies to manage non-SNMP devices.
- Configure a specific node or a group of nodes to have different values than the default configuration.

In a distributed network environment, make sure that you configure managed network devices to enable SNMP communication from client machines as well as from the manager workstation. If you do not properly configure the network device, all SNMP requests from the clients, such as MIB applications, MIB browsers, and so forth, will time out with no response from the device.

How an Application Uses SNMP Configuration

When an application initiates an SNMP request, the SNMP APIs look for a configuration entry in the node list to query the node. If the application cannot find an entry, it looks for the first IP address wildcard entry in the network list that matches the IP address of the node. If an entry exists, then it's used. If a network list entry does not exist for the node, the application uses the default SNMP configuration.

In a distributed network environment, SNMP requests issued from the client are directly transmitted on the network from the client. That is, the requests do not go through the server first. Therefore, clients must have their community names configured to enable SNMP requests on a node.

Description of the SNMP Configuration Dialog Box

Figure 30 on page 168 shows the SNMP Configuration dialog box.

Figure 30. SNMP Configuration Dialog Box

The SNMP Configuration dialog box contains two main sections. The top section contains three selection lists that display current SNMP configuration values for:

- Specific nodes
- A group of nodes
- The default SNMP configuration for your network

The bottom section contains text entry fields in which you can specify new values for entries in the top section.

Note: If you change the value for configuration checking to a shorter interval, you must perform a demand poll on a node to make that change effective. Otherwise, you would need either to wait 24 hours or to update the rate of the daily configuration check.

When you select an item from any of the three selection lists in the top section, the current parameters for that entry are displayed in the bottom section. You can then view, delete, or modify the parameters.

The order of the three selection lists, and the order of entries within each list, illustrates the precedence order in which entries are searched when a node is queried.

Note: The parameters displayed in the selection lists can be tailored by the X11 resource xnmsnmpconf.summaryList. If this resource is set to True, only the

Target, Community, Set Community and Proxy parameters are displayed in the selection lists. Otherwise, all parameters are displayed in the selection lists.

Steps

The values you enter for individual nodes override the values set in the Topology/Status Polling Configuration dialog box. The Enable Polling and Discovery Settings button must be selected in that dialog box for the values you set in the SNMP Configuration dialog box to apply. To change the configuration for an SNMP node, follow these steps:

Step 1. Select **Options..SNMP Configuration** from the main menu.

Step 2. Select the item you want to change from one of the selection lists. The configuration for the item you selected is displayed in the SNMP Parameters section.

Step 3. Change the following configuration values as appropriate.

- Proxy
- Target
- Community
- Set Community
- Timeout
- Retry count
- Remote Port
- Status Polling

Select the **Help** button for information about these fields.

Step 4. Select the **Replace** button to display the new values in the appropriate selection list.

Step 5. Select the **OK** or **APPLY** button to apply the changes.

To configure a specific node or a group of nodes to have different configuration values than the default configuration, follow these steps:

Step 1. Select **Options..SNMP Configuration** from the main menu.

Step 2. Enter the values in the text entry fields in the SNMP Parameters section. If you specify a specific node, you only need to fill in the fields that have different configuration values than the default configuration. If you specify a group of nodes using an IP address wildcard, all the fields must be filled in.

Step 3. Select the **Add** button to add the new values to the appropriate selection list. If the target is a regular host name or an IP address, the configuration is added to the Specific Nodes selection list. If the target is an IP address with wildcards, the configuration is added to the IP Address Wildcards selection list.

Step 4. Select either the **Apply** or **OK** button to apply the changes.

If you can't set the isSNMPProxied general attribute in the map database, add the target to the map. The target must exist in the map database and have a selection name.

SNMP Configuration Dialog Box Fields

The following fields appear in the SNMP Configuration dialog box:

Specific Nodes selection list
: Displays the current SNMP configuration for all individual nodes. To change the values for an entry, select the entry in the list and change the appropriate values in the fields in the SNMP Parameters (bottom) section of the dialog box.

IP Address Wildcards selection list
: Displays the current SNMP configuration for specified groups of nodes within a network. These networks are specified using IP address wildcards, for example, *.*.*.5. To change the values for an entry, select the entry in the list and change the appropriate values in the fields in the SNMP Parameters (bottom) section of the dialog box.

  To control which values SNMP will use when an SNMP request matches multiple wildcard IP addresses, use the **Reorder** button. SNMP will use the first entry in the list that matches the IP address used in the SNMP request. To change the order of the entries, select the item you want to reorder from the selection list and select the up or down arrows on the **Reorder** button. Each click moves the item (or items) up or down one position.

Default selection list
: Displays the default SNMP configuration for your network. To change the default values, select the entry in the list and change the appropriate values in the fields in the NetView SNMP Parameters (bottom) section of the dialog box.

Proxy
: Selectable and read-write field. Select the Use Proxy to access Target check button to specify whether or not the SNMP target is queried through the proxy node. If you select this check button, the Proxy field becomes available. Enter a proxy name in this field. Valid values include either a host name or an IP address. If you do not select this check button, the Proxy field remains grayed.

  When you specify a proxy, the general attribute isSNMPProxied for the target node is set to true. If you later modify the configuration entry to disable use of a proxy for the target node, the isSNMPProxied attribute is reset to false.

Target
: Read-write field. Enter the address of the destination managed node or the group of nodes to be queried. The target can be any of the following:

  - A regular host name

- An IP address
- An IP address with wildcards
- A non-SNMP node that has a proxy

The program automatically adds the target to the appropriate selection list in the top section of the dialog box.

Community

Read-write field. Enter the SNMP community name that the management application uses as authorization when issuing SNMP Get and GetNext requests to the target node. This community name is also used for SNMP Set requests if the management application cannot resolve an explicit Set Community name for the Target node. Valid values include any character that you can enter from the keyboard, including spaces.

Set Community

Read-write field. Enter the SNMP community name that the management application uses as authorization when issuing SNMP Set requests to the target node.

SNMP agents are often configured with different community names for Set requests than for Get requests. This field enables you to configure the SNMP management applications according to this distinction. Valid values include any character that you can enter from the keyboard, including spaces.

If you leave the Set Community field blank, the value in the Community field is used for both Set requests and Get requests.

Timeout

Read-write field. Enter a value indicating the amount of time (in seconds) that the management application will wait for a response before attempting to retry the SNMP request to the target node. Valid values include 0.1 to 99; the timeout value cannot be 0. If you need a larger maximum value, you can change the upper limit by increasing the value of the X11 resource xnmsnmpconf.maxTimeout.

If SNMP requests to all agents are timing out, increase the default timeout value. If SNMP requests to a specific agent are timing out, configure a different timeout value for that agent.

If you configure a longer time-out interval, less traffic will be generated. However, if your time-out interval is set too large, you might not be notified of problems as quickly as you want to be notified.

Retry Count

Read-write field. Enter a value indicating the maximum number of times the management application will attempt to make before concluding that the target node is unreachable. Valid values include 0 – 99. If you need a larger maximum value, you can change the upper limit by increasing the value of the X11 resource xnmsnmpconf.maxRetry.

Remote Port

Read-write field. Enter the UDP port number on the target or proxy node (as appropriate) where the SNMP agent expects to receive SNMP requests. The standard SNMP port is 161. This field is generally used only for specialized proxy agents that do not listen to the standard SNMP port.

Status Polling

Read-write field. Enter a number followed by an s, m, h, d, or y to specify seconds, minutes, hours, days, or years, indicating the frequency at which the netmon daemon uses ping to query the status of a given target. This is the amount of time it takes for a symbol to change colors if its status changes. The default is five minutes. Note that the more frequently you poll for status, the more ping traffic the manager and networks have.

Configuring a Backup Manager

With NetView for AIX, you can segment a large network and create individualized spheres of control for several management stations. Objects outside an operator's sphere of control are unmanaged by that operator's management station. When an object is unmanaged, NetView for AIX no longer polls that object for status and configuration changes. You can have numerous NetView for AIX programs on the network. Each one can be configured so that there is little duplication of management network traffic.

Using the Backup Configuration dialog box, you can configure objects as manager nodes (NetView for AIX programs) or as managed containers, and then specify which manager is managing each container. A *managed container* is a collection of objects and is designated as being managed by one or more managers or not managed at all.

The entire network can be separated into various containers using the concept of a partitioned Internet map. Each manager can be designated to manage a subset of the containers. Managed containers that are defined and managed by a remote NetView for AIX managing system are unmanaged.

Each manager checks on the status of the other known managers and notifies the backup manager when a manager is disabled or when the manager is restored. The operator can then manage the backup session. See “Managing a Backup Session” on page 180 for more detailed information. Depending on the configuration of the local manager, it can manage all or part of the disabled manager's containers.

A submap labeled ManagerSubmap is displayed on the root submap. The Manager submap contains the symbol for each manager node that is discovered on the network. The Manager submap's only function is to group all managers together in one place, allowing you to quickly determine current backup sessions for the selected remote manager. See “Determining Current Backup Sessions” on page 181 for more detailed information.

Configuring Manager-Container Associations

To configure nodes as managers and specify the containers they manage, do one of the following:

- Create a seed file that specifies which associations to make
- Use the Backup Configuration dialog box

There is no manager-to-manager communication. Therefore, to configure manager-container associations across multiple managers, create a definitive seed file and distribute it to all managers. This ensures the same configuration file is used for all managers. If you change a map or a manager-container association on a manager, change all managers. The manager-container configurations are global only on the local system. Use the Backup Configuration dialog box to display and modify the associations already configured.

Manager-Container Characteristics

Managers and containers must have the following characteristics to be valid:

- The name specified must be the selection name of an object in the object database.
- The specified manager name is a node (the *isNode* capability must be set to *True*).
- The specified container name must be a Location, Internet, Network, or Segment object.

Using a Seed File to Configure Managers

You can create a seed file to pass to the backup process. A seed file is used by the NetView for AIX program to determine which nodes are managers, as well as how manager-container associations should be made. It defines the relationship between managers, containers, and backup managers. The seed file is passed to the backup process by modifying the command line for the backup process in the application registration file, /usr/OV/registration/C/backup. Find the line:

```
Command -Shared -Initial -Restart "${BackupDir:-/usr/OV/bin}/backup";
```

Add the seed file name to the end of the line as follows:

```
backup [-s /path/seed_file_name]";
```

Save the seed file in the /usr/OV/conf directory. Include the full path name. Invoke the seed file through the registration file and not from the command line.

Seed File Format: The seed file consists of three fields per line:

```
active manager    container    backup manager
```

Where:

`active manager` Is the name of a manager. This is the active manager for the container object.

`container` Is the container object name.

`backup manager` Is the name of the backup manager.

You can designate more than one backup manager as shown in the following example. Names must be enclosed in double quotation marks:

```
"bos.rtp.ibm.com"   "192.10.30"  "rudd.rtp.ibm.com"  "cowan.rtp.ibm.com"
```

Where:

`bos.rtp.ibm.com` Is the manager node.

`192.10.30` Is the container node.

`rudd.rtp.ibm.com` and `cowan.rtp.ibm.com` Are backup manager nodes.

If `bos` should go down, both backup managers will start managing the specified container objects.

Seed File Process: The seed file is processed line by line. The manager name and the container name are read. These objects are validated to make sure that they actually exist and that they have the appropriate characteristics. If the objects are valid, the manager object's capability field, isManager, is set to *True*. Associations between the manager and container are activated. The container object is added to the manager's selected containers list and the manager object is added to the container's selected managers list. The container is checked to see if it should appear as managed or unmanaged by the ipmap application. After those associations are made, the associations between the backup managers and the containers are made.

If the seed file has been updated and changed or the necessary managers and containers were not discovered when the seed file was initially read, select **Administer..Backup..Read Seedfile** from the NetView for AIX main menu to re-read the seed file. This option adds associations to those that already exist. Associations are not deleted.

Seed File Example

In the following example, M1, M2, and M3 are manager nodes in a network. Each manager has the following backup seed file configured:

```
"M1"     "Backbone"                  "M2"
"M1"     "North_America_Network"     "M2"
"M1"     "Latin_America_Network"     "M3"
"M1"     "South_America_Network"     "M3"
"M1"     "Antarctica_Network"
"M2"     "Africa_Network"            "M1" "M3"
"M2"     "Eastern_Europe_Network"    "M1"
"M2"     "Western_Europe_Network"    "M1"
"M3"     "Asia_Network"              "M1" "M2"
"M3"     "Australia_Network"         "M1"
"M3"     "Middle_East_Network"       "M1" "M2"
```

When M1 is started, Backbone, North_America_Network, Latin_America_Network, South_America_Network, and Antarctica_Network are automatically managed.

Africa_Network, Eastern_Europe_Network, Western_Europe_Network, Asia_Network, Australia_Network, and Middle_East_Network are automatically unmanaged.

When M2 is started, Africa_Network, Eastern_Europe_Network, and Western_Europe_Network are automatically managed. All the other containers are unmanaged.

When M3 is started, Asia_Network, Australia_Network, and the Middle_East_Network are automatically managed. All the other containers are unmanaged.

In this example, each manager has a subset of the whole network to manage. M1 is acting as the backup for all containers of M2 and M3. M2 and M3 are backing up specific containers. The Antarctica_Network is not backed up at all.

Using the Backup Configuration Dialog Box

Use the Backup Configuration dialog box, shown in Figure 31 on page 176, to designate nodes as managers and add, display, and modify the configuration of backup managers and containers. Define and configure remote manager nodes on the local manager node so they can be backed up.

The Backup Configuration dialog box displays a list of all configured manager nodes (nodes that have the isManager capability field set to *True*) and a list of all the container nodes known by NetView for AIX.

You must have a read-write map to configure backup managers. Only one Backup Configuration dialog box can be opened for each object database.

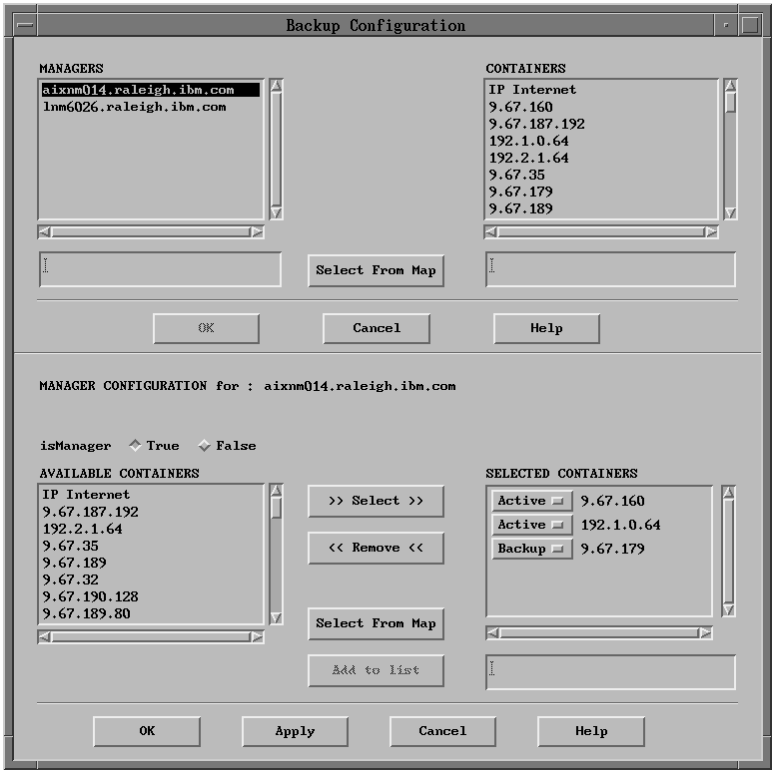


Figure 31. The Backup Configuration Dialog Box

Designating Nodes as Managers: To designate nodes as managers, follow these steps:

Step 1. Select **Backup..Backup Configuration** from the Administer pull-down menu. The Backup Configuration dialog box is displayed.

Step 2. Enter the name of the node in the Selection field or select an object on the submap and click on **Select From Map**. When you select an object from the map, the object characteristics are checked. If it is a valid manager, its selection name is placed in the manager selection box.

Step 3. Select **OK**. The Manager Configuration pane is displayed.

Step 4. Select the **True** button next to the isManager field. The manager node name is added to the list of managers in the Managers field. When using a seed file, the isManager field is automatically set to *True*.

Step 5. Select **OK** or **Apply**.

Step 6. Repeat the steps for each node you want to designate as a manager.

Once you have designated nodes to be managers, you can make your manager-container associations.

Modifications to the isManager capability field causes NetView for AIX to re-evaluate the managed or unmanaged state of all containers in the Selected Containers list.

Adding Manager-Container Associations: Configure on the local managing system the remote managers and which containers they are actively managing. Also add the containers you want the local manager to backup if the active manager for those containers should go down. The manager-container associations that you configure apply to the current system's database only.

You can make a manager-container association even though the selected node does not have the isManager capability set to true.

Dividing the backup management of containers between managers prevents a sphere of control, which could be an entire network, from being placed on one manager.

Steps: To configure manager-container associations, follow these steps:

Step 1. Select **Backup..Backup Configuration** from the Administer pull-down menu. The Backup Configuration dialog box is displayed.

Step 2. Do one of the following:

- Select a Manager or a Container from either list.
- Enter the name of the node in the Selection field.
- Select an object on the submap and click on the **Select From Map** button. When you select an object from the submap, the object characteristics are checked. If it is a valid manager or container, its selection name is placed in the appropriate selection box.

Step 3. Select **OK** if you select an item from the list.

If a manager was selected:

a. The Manager Configuration pane is displayed.

b. Do one of the following:

- Select a container from the Available Containers list. This list displays all of the available containers that are not associated to the selected manager. Click on **Select** to create an association between the manager and the selected container. The container name is removed from the Available list and added to the Selected Containers list.
- Type a container object name in the field below the Selected Container list and select the **Add to List** button. The container name is added to the Selected Containers list.
- Select an object from the map, click on **Select from Map**, then select **Add to List**. The container name is added to the Selected Containers list.

The Selected Containers list displays all of the containers associated with the selected manager. Selecting containers from the Selected Con-

tainers list can affect the managed or unmanaged state of that container on the manager.

c. Select the type of association to be made for the selected manager and container.

   Select **Active** if you are configuring associations between a manager and the containers that the manager is managing. There can be more than one manager configured as active for each container.

   Select **Backup** if you are configuring associations between a manager and the containers that this manager is going to manage if the active manager should go down. Multiple backup managers for a single container and multiple backup containers per backup manager are permitted.

d. Repeat the steps for each manager-container association.

If a container was selected:

a. The Container Configuration pane is displayed.

b. Do one of the following:

   - Select a manager from the Available Managers list. This list displays all of the available managers that are not associated with the selected container. Click on **Select** to create an association between the container and the selected manager. The manager name is removed from the Available list and added to the Selected Managers list, which displays all of the managers associated with the selected container.
   - Type a manager name in the field below the Selected Managers list and select the **Add to List** button. The manager name is added to the Selected Managers list.
   - Select an object from the map, click on the Select from Map button and then the **Add to List** button. The manager name is added to the Selected Managers list.

c. Select the type of association to be made (active or backup).

d. Repeat the steps for each manager-container association you want to add.

Step 4. Select **Apply** to activate your associations.

Step 5. Select **OK** to close the Container Configuration pane.

Deleting Associations: Associations can be deleted by selecting a container from the Selected list and selecting the **Remove** button. Deleting containers from the selected list can affect the managed or unmanaged state of that container on the local manager.

To delete managers from the Managers list, select a manager and select **False** next to the isManager capability field.

If you remove associations that contain non-manager nodes, the manager is moved from the Selected Managers to the Available Managers list when you select Remove. If you exit the Container Configuration pane and then return, the node is no longer on the Available Managers list because the capability was not set to isManager.

Determining Remote Manager Status

The status of remote managers is determined by Node Up and Node Down traps sent by netmon to the local manager.

When the local manager receives a Node Down trap for a remote manager, all active containers for that remote manager are examined. If a container is also part of the local manager's sphere of control, no further action is taken because the local operator already has control of that container. By default, the local manager receives notification of a disabled manager. A Manager Down message box is displayed, and the operator can initiate actions from the message box to manage the backup session.

When the local manager receives a Node Up trap for a remote manager, a Manager Restored message box is displayed stating the remote manager's name and the fact that it is now active. The operator can initiate actions from the message dialog box to manage the backup session. No further action is taken by the backup process until the operator returns the container to unmanaged state by closing the container submap from the Navigation Tree window, the Manager Submap, or through the Manager Restored message box. At that time the managed or unmanaged state of the container is re-evaluated.

Note: Unmanaging remote managers or submaps and containers in which the remote manager objects reside prevents the local manager from being notified when the remote manager is disabled, because the local manager no longer polls those objects.

Manage-Unmanage Rules

When the configuration changes for a container, for example, a container is removed from the list of Selected Containers associated with a manager, the NetView for AIX program determines whether to automatically manage or unmanage the container.

When the configuration changes for any manager, for example, the isManager capability changes from true to false, the managed or unmanaged condition for all of that node's managed containers is updated. Here is how NetView for AIX handles status changes:

- If the local manager and the given container are associated, the container is managed.
- If the container is not associated with any manager, the container is managed by the local manager.
- If an association exists between this container and any remote node where the isManager capability is set to True, the container is unmanaged by the local manager.

- If all remote associations are with nodes where isManager is set to False, the container is managed by the local manager.

Managing a Backup Session

This section describes the actions you can take when you receive notification of a remote manager status change. This section also describes how you can customize the backup default actions, and how you can determine what containers a backup manager is managing.

Responding To a Manager Down Notification

When a remote manager is disabled, a Manager Down message box is displayed. You can perform one of the following actions:

- Select the **OK** button to manage all the backup containers and open submaps for those containers.
- Select the **Manage** button to manage all the backup containers but do *not* open submaps for those containers.
- Select the **Cancel** button to take no action.

Responding To a Manager Up Notification

When a remote manager is restored, a Manager Restored message box is displayed. You can perform one of the following actions:

- Select the **OK** button to take no action and close the message box. This gives the operator the opportunity to complete tasks being performed in the disabled manager's submaps. When the tasks are completed, the submaps must be closed from the Navigation Tree or from the Manager Submap to make sure that the containers are automatically unmanaged.
- Select the **Close All** button to unmanage all the backup containers and close any submaps for those containers that are open.

Changing Backup Default Actions

You might want to change backup default actions to prevent notification of remote manager status changes and to automatically perform a specified action. You can change entries in the /usr/OV/app-defaults/Backup file or copy the appropriate entries in the user's .Xdefaults file to change backup default actions.

The following lines in the /usr/OV/app-defaults/Backup file contain the default settings for the actions taken for remote manager status changes:

```
backup*ManagerDown.displayPopup:    TRUE
backup*ManagerDown.defaultAction:   OK

backup*ManagerUp.displayPopup:      TRUE
backup*ManagerUp.defaultAction:     OK
```

You can set both displayPopup values to `FALSE` to prevent notification of remote manager status changes. The values for ManagerDown.defaultAction (OK, Manage, or

Cancel) and ManagerUp.defaultAction (OK or Close All) correspond to the actions initiated from the Manager Down and Manager Restored message boxes. See "Responding To a Manager Down Notification" and "Responding To a Manager Up Notification" for a description of those actions.

To automatically manage and unmanage backup containers without displaying notification of remote manager status, do the following:

- Change both displayPopup values to `FALSE`.
- Change the ManagerDown.defaultAction to `Manage`.
- Change the ManagerUp.defaultAction to `Close All`.

If you make changes to any of the app-defaults files after NetView for AIX is started, use the command `xrdb -merge .Xdefaults` to load the new resource. If you have not started NetView for AIX, the new resource will be loaded when you start the graphical interface.

Determining Current Backup Sessions

You can determine what containers a selected manager is managing on behalf of another manager using the Backup Session window. From the Backup Session window, you can stop managing some or all of the backup containers, and you can open submaps for selected containers. To determine current backup sessions, follow these steps:

Step 1. Select a manager symbol from either the Manager submap or another submap that contains a manager symbol.

Step 2. Select **Administer..Backup..Display Sessions...** from the context menu. The Backup Session Window is displayed.

Step 3. Do one of the following:

- Select specific containers from the list and select the **Close** button to stop managing some, but not all, of the containers in the list.

  The selected containers will be unmanaged, and any open submaps for those containers are closed.

- Select the **Close All** button to stop managing all the containers in the list.

  All containers in the list will be unmanaged, and any open submaps for those containers are closed.

- Select specific contains in the list and select the **Open Submap** button to open the submap for one or more containers in the list.

  Submaps are opened for those containers. If a submap is already open for any selected container, it is placed on top of the other windows that are displayed.

- Select the **Cancel** button to exit the session window without performing any action.

Chapter 7. Managing Network Performance

The NetView for AIX program helps you manage network performance by providing several ways to track and collect performance information for objects on the network. You can use performance information to help you:

- Monitor the network for signs of potential problems
- Resolve network problems
- Collect information for trend analysis
- Generate regular performance reports
- Allocate network resources
- Plan future resource acquisitions

This chapter describes MIBs and explains how to load, unload, and browse them. It also describes how to use the NetView for AIX program's predefined performance applications and how to create your own applications to monitor network performance. The following topics are covered in this chapter:

Loading and Unloading MIBs

The Options..Load/Unload MIBs and Options..Load/Unload MIBs..SNMPv1/SNMPv2 operations lets you include your enterprise-specific MIB, or the enterprise-specific MIB for a device that you use on your network, in the NetView for AIX program's MIB description file. These menu operations work for SNMPv1 MIBs and SNMPv1 or SNMPv2 MIBs, respectively. You can also use these operations to load other MIBs.

The purpose of loading a MIB is to define the MIB objects so the NetView for AIX program's applications can use those MIB definitions. The MIB you are interested in must be loaded on the system where you want to use the MIB Browser, MIB Data Collector, MIB Application Builder, and the applications built by the MIB Application Builder. In a distributed network environment, load enterprise-specific MIBs on the NetView for AIX server. Because the directory where the MIBs are stored, /usr/OV/conf, is automatically NFS mounted onto the client machines during the NetView for AIX installation process, loaded MIBs are available to the clients. If you have unloaded all the MIBs in the MIB description file, you must load MIB-I or MIB-II before you can load any enterprise-specific MIBs.

Loading MIBs

To load an enterprise-specific MIB, you can first copy the MIB into the default directory, /usr/OV/snmp_mibs. Otherwise, if you know the full path name of the directory where the MIB is located, you can enter it in the MIB File to Load text field in the Load MIB From File dialog box.

Each MIB that you load adds a subtree to the MIB tree structure. You must load MIBs in order of their interdependencies. A MIB is dependent on another MIB if its highest node is defined in the other MIB. For example, the MIB ibm-alert.mib is dependent on the MIB ibm.mib, so ibm.mib must be loaded before ibm-alert.mib is loaded.

Steps

To load a MIB, take the following steps:

Step 1. From the NetView for AIX main menu, select either **Options..Load/Unload MIBs..SNMP...** for SNMPv1 MIBs or **Options..Load/Unload Mibs..SNMPv1/SNMPv2...** for SNMPv1 or SNMPv2 MIBs.

The Load/Unload MIBs window is displayed, containing a scrollable list of all loaded MIBs, as shown in Figure 32.

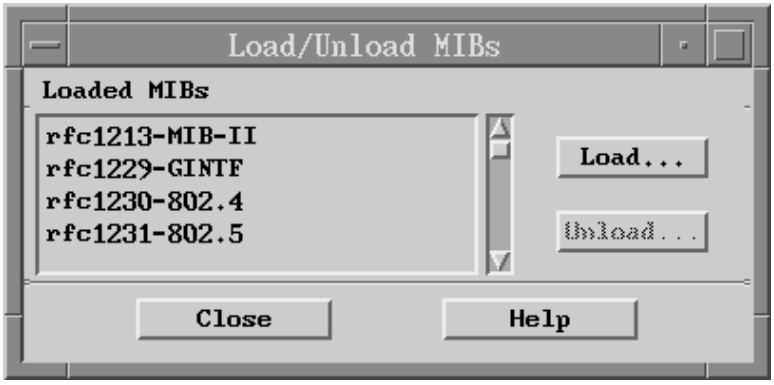


Figure 32. Load/Unload MIBs Dialog Box

Step 2. Select **Load...** to display the Load MIB From File dialog box, which contains a scrollable list of the MIB files in the default MIB directory, /usr/OV/snmp_mibs.

Step 3. Select the MIB you want to load.

Step 4. Select **OK**.

The Load/Unload MIBs operation stores loaded SNMPv1 MIBs in the /usr/OV/conf/snmpmib database and SNMPv2 MIBs in the /usr/OV/conf.snmpv2mib database, which is known as the Loaded MIB Database. Do not try to edit this file directly; instead, make any changes to this file through the Load/Unload MIBs operation.

Once the MIB is loaded, you can traverse the MIB tree and select objects from the enterprise-specific MIB to use in the following operations:

- Browsing MIBs
- Collecting MIB data
- Building MIB applications
- Running applications you build with the MIB Application Builder

Unloading a MIB

To unload enterprise-specific MIBs, select the MIB from the list of loaded MIBs in the MIB Load/Unload MIBs window and select **Unload...**. Select **OK** in the Unload MIB—Confirmation dialog box to unload the MIB.

Browsing MIBs

Use the MIB Browser to query and set MIB values for both Internet-standard and enterprise-specific MIB objects.

You can also use the MIB Browser to graph MIB objects and their specific instances. Some MIB objects can occur several times per network object, each time with a different value. Each such occurrence of the MIB object is called an *instance*.

For example, the interfaces MIB object ifDescription has as many instances as its associated network object has interfaces, as shown in the following example:

```
1 : lo0; Software Loopback
2 : tk0; trty0; IBM 6611 Token-Ring Network Interface
3 : tk1; trty1; IBM 6611 Token-Ring Network Interface
```

However, the system MIB object sysContact has only one instance per network object, because generally there is only one person so designated. This is MIB instance 0, as shown in the following example:

```
sysContact.0: J. J. Wanscott 555-1234
```

To use the MIB Browser, follow these steps:

Step 1. Select an SNMP network object whose MIB objects you want to view on a submap.

Step 2. From the NetView for AIX main menu, select either **Tools..MIB Browser..SNMP...** for SNMPv1 agents or **Tools..MIB Browser..SNMPv1/SNMPv2...** for SNMPv1, SNMPv2, or SNMPv2USEC agents.

The Browse MIB window contains the name or IP address of the network object you selected, as shown in Figure 33 on page 186.

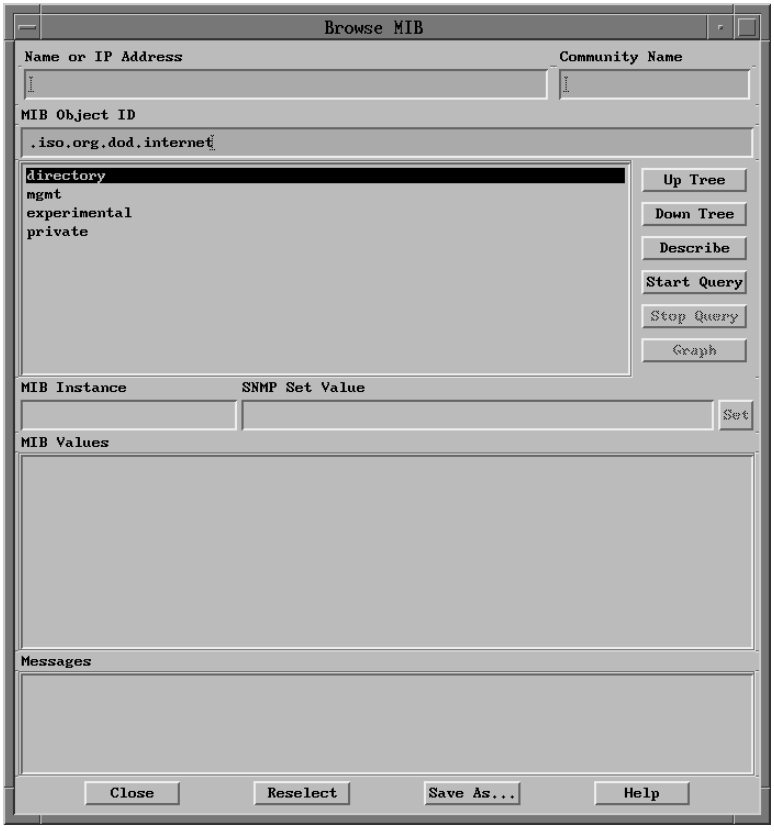


Figure 33. Browse MIB Dialog Box

Step 3. In the Community Name field, enter the community name of the agent that is running on the selected object. If left blank, the community name defaults to public.

The community name functions as a password for different levels of access to MIB objects. For example, to retrieve the value of a MIB object, you need to know the community name that permits SNMP Get operations. Generally, this community name defaults to `public`. If you want to change the value of a MIB object, you must know the community name that permits SNMP Set operations.

Step 4. Select nodes along the path of the MIB tree and select either Up Tree or Down Tree to traverse the tree. Generally, the MIB objects you will be working with are contained in either the `mgmt` branch, which contains standard MIB definitions, or the `private` branch, which contains enterprise-specific MIB definitions.

Step 5. Once you have reached a leaf node, or actual MIB object, select Describe to see a description of the object to determine whether it is the one you want.

The Describe MIB Variable dialog box displays the name of the MIB object, its object ID, and the type of MIB object it is, and gives a brief description of its meaning.

Figure 34 shows the description of the snmpOutTraps MIB object.

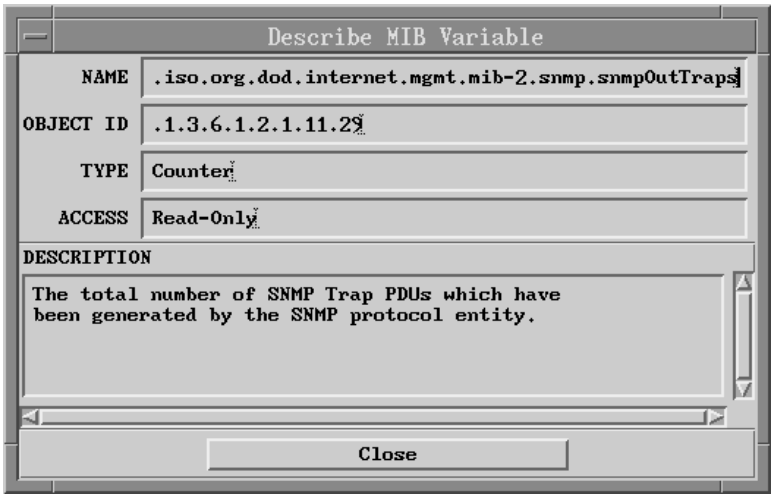


Figure 34. MIB Object Description—snmpOutTraps

Step 6. To query the MIB object for its current value, select **Start Query**.

You do not have to be at a leaf node in order to use the Start Query operation. You can begin a query at any intermediate node to retrieve all the MIB objects in that node's subtree at the same time. Once you see the list displayed in the MIB Values field, you can select one and look at its description. The current value of the MIB objects will be displayed in the MIB values fields. To stop the query, select the **Stop Query** button.

Step 7. You can graph real-time data on the MIB object you have selected, provided it is numeric. Select the **Graph** button to display the NetView for AIX Grapher window and look at the real-time values of the selected MIB object.

Using the NetView for AIX Performance Applications

The NetView for AIX program provides applications that enable you to monitor both real-time and historical network performance. The predefined applications monitor and graph real-time network performance. These predefined applications are described in “Using NetView for AIX Predefined Applications” on page 189. The MIB Application Builder enables you to create your own applications to collect, display, and save real-time MIB data.

The MIB Data Collector provides a way to collect and analyze historical MIB data over long periods of time to give you a more complete picture of your network's performance. The MIB Application Builder is described in “Building MIB Applications” on page 192.

Comparison of MIB Applications

Table 14 compares the MIB Application Builder and the MIB Data Collector.

Table 14 (Page 1 of 2). Comparison of NetView for AIX MIB Applications

Task	MIB Application Builder	MIB Data Collector
Starting	Must always be started by selecting **Tools..MIB Application Builder** from the main menu	Data collection automatically started when the NetView for AIX program is started, unless the collection is in suspended mode
Selecting MIB Variables	• Can select more than one MIB variable • Can apply selection rules to MIB variables	• Can collect data for one or all MIB instances • Can specify a single interface, but data is collected on the MIB index (instance)
Specifying Polling Intervals	Can specify polling intervals only for Graph format	Can change polling intervals for individual devices
Selecting Agents	Agents must be selected from map	Define agents by using pattern-matching characters, selecting from map, or by entering the agent name in the field.
Setting Thresholds	Not applicable	Can choose one of the following: Exclude Collection Store, Check Thresholds Store, No Thresholds Don't Store, Check Thresholds
Generating Events	Not applicable	Can generate events and define specific trap numbers
Selecting Output Format	Choose table, form, or graph	Shows data in table format. You can select **Graph** from the table display.
Customizing Output	• Can customize the “name” of the MIB variable displayed in the title bar • Cannot customize line names • Customization of graphic display lost between invocations	Uses applications in the /usr/OV/reports/C/ directory to graph output and generate reports
Saving Output	Can save output to /usr/tmp/xnmappmon.save or choose another file name	Collection file always stored in /usr/OV/database/snmpCollect. You cannot specify the file name
Exporting to ASCII Format	• Cannot export graph data to ASCII format • Can export summary instance table format • No time points are provided	• Can convert to ASCII format • Time points provided through SMIT

Table 14 (Page 2 of 2). Comparison of NetView for AIX MIB Applications

Task	MIB Application Builder	MIB Data Collector
Performing Other Calculations on MIB Variables	Not applicable	Apply expressions in /usr/OV/conf/mibExpr.conf file to MIB variables

Monitoring Real-Time Network Performance

The NetView for AIX program provides applications that enable you to monitor your network's real-time performance. It also provides you with a tool, the MIB Application Builder, that enables you to build your own applications. This section describes the NetView for AIX program's predefined applications and the MIB Application Builder.

Using NetView for AIX Predefined Applications

Many of the performance applications described in this section display their data in graphs. Some performance applications do not graph the data they collect; see the individual application description for more information. Refer to the Help system entries for each application for complete information.

Monitoring CPU Performance: If you suspect that some network nodes are handling more tasks than others and might become overloaded, you can check this by selecting **Monitor..System Activity..CPU Performance** from the NetView for AIX main menu. To use this application, one of the following conditions must be true:

- The trapgend daemon must be installed on a RISC System/6000* node.
- The node must be an HP node.
- The Systems Monitor feature of the NetView for AIX program must be installed.

This application monitors and graphs the average number of jobs performed in the last minute. Use this application to help determine if the system has workload balancing problems.

Monitoring Disk Space: You can check the status of disk space on a remote SNMP agent node by selecting the **Monitor..System Activity..Disk Space** operation. This operation provides the following information:

- Name of the file system
- Total KB of space
- Number of KB used
- Number of KB available
- Percent of total capacity being used
- Directory name on which the file system is mounted

To use this application, one of the following conditions must be true:

- The trapgend daemon must be installed on a RISC System/6000 node.
- The node must be an HP node.
- The Systems Monitor feature of the NetView for AIX program must be installed.

Monitoring Interface Traffic: If you suspect network performance problems, you can select **Monitor..Network Activity..Interface: Traffic** from the NetView for AIX main menu to see a graph of packet statistics for selected SNMP nodes. If the graph reveals that packets are not being received or transmitted by certain nodes, you can then begin fault management procedures for those nodes.

Monitoring Ethernet Performance: You can monitor and graph performance for Ethernet networks by using the operations listed under **Monitor..Network Activity..Ethernet** on the NetView for AIX main menu.

Monitoring Ethernet Traffic: The **Monitor..Network Activity..Ethernet..Traffic** operation monitors and graphs interface statistics for a Cisco gateway server that supports SNMP. The following information is displayed in the graph:

- The number of packets successfully transmitted by the interface hardware
- The number of packets successfully received by the interface hardware
- The number of bits per second successfully transmitted
- The number of bits per second successfully received

Monitoring Ethernet Errors: To view interface error statistics for a gateway server that supports SNMP, select the **Monitor..Network Activity..Ethernet..Errors** operation from the NetView for AIX main menu. This operation graphs the average number per second as computed over the previous polling interval for the following statistics:

- The number of packets received with a Cyclic Redundancy Checksum (CRC) value that is not valid
- The number of packets received with a Frame Check Sequence (FCS) value that is not valid
- The number of packets received that are smaller than the physical media permits
- The number of packets received that are larger than the physical media permits
- The number of packets that arrive too quickly for the interface hardware to receive.

Monitoring TCP Connections: The **Monitor..Network Activity..TCP: Connections** operation lists the TCP connection table for selected remote SNMP nodes. It shows the local and remote addresses of the node and the state of each session. This operation can help you determine whether performance problems are caused by the TCP layer instead of the SNMP layer.

Monitoring SNMP Network Activity: You can monitor and graph performance for SNMP networks by using the operations listed under **Monitor..Network Activity..SNMP** on the NetView for AIX main menu.

Monitoring SNMP Traffic: Use the **Monitor..Network Activity..SNMP..Traffic** operation to monitor and graph the SNMP network traffic to and from selected nodes. You can compare this graph to general interface traffic graphs to determine the percentage of traffic to and from the selected node that is attributed to the SNMP management of the selected node. The following information is displayed in the graph:

- The number of packets per second received by the selected node that carry SNMP traffic
- The number of packets per second transmitted by the selected node that carry SNMP traffic

Monitoring SNMP Operations: Use the **Monitor..Network Activity..SNMP..Operations** operation to monitor and graph the SNMP operations requested of and performed by the SNMP agent on the selected nodes. The number of operations reported is the total number of operations for the given category that have occurred since the SNMP agent on the selected node was last started. The following information is displayed in the graph:

- The total number of MIB objects retrieved successfully from the selected node through Get and Get Next requests
- The total number of MIB objects modified successfully on the selected node through Set requests
- The number of SNMP Get requests received and successfully processed by the selected node
- The number of SNMP Get Next requests received and successfully processed by the selected node
- The number of SNMP Set requests received and successfully processed by the selected node
- The number of SNMP Traps (unsolicited notifications) transmitted by the selected node

Monitoring SNMP Errors: Use the **Monitor..Network Activity..SNMP..Errors** operation to display a summary of the SNMP protocol errors detected by the SNMP Agent on the selected nodes. The number of errors reported is the total number of errors for the given category that have occurred since the SNMP Agent on the selected node was last started. The following information is displayed in the graph:

- ASN parsing errors
- SNMP requests by an unsupported version of SNMP
- SNMP requests received with an incorrect community name
- SNMP requests received with a community name that is inappropriate for the operation requested
- SNMP requests that could not be processed because the request message is larger than the maximum message size supported by the selected node
- SNMP requests that contained a MIB object not supported by the selected node
- SNMP set requests that contained values not valid for the specified MIB object
- SNMP requests that could not be processed for some other reason

Monitoring SNMP Authentication Failures: An authentication failure occurs when a management system sends an SNMP request to an agent but does not send the correct community name, or password, with the request.

If you have a map open with read-write access, you can obtain a list of the management systems that have caused authentication failures on a selected node. The node you select must be an HP node.

Select the **Monitor..Network Activity..SNMP..Authentication Failures** option from the NetView for AIX main menu. From the resulting display, you can learn that:

- Certain management systems are not permitted to communicate with this particular agent.
- The agent might have a configuration problem, which you can take steps to resolve.

Building MIB Applications

The MIB Application Builder enables you to build MIB applications without programming. This is helpful when you want to monitor real-time performance of specific MIB objects. You can build MIB applications that poll certain MIB objects on a regular basis and produce output such as forms, tables, or graphs.

The new MIB application is placed in the Monitor menu by default. To look at MIB values returned by the new application, select the appropriate item under the Monitor option on the NetView for AIX main menu.

Select **Tools..MIB Application Builder** from the NetView for AIX main menu. The MIB Application Builder dialog box is displayed, as shown in Figure 35 on page 193.

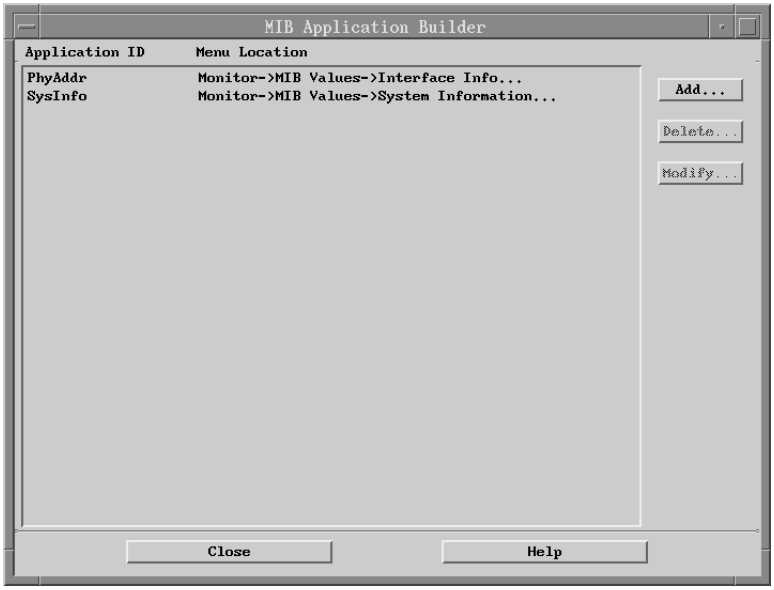


Figure 35. MIB Application Builder Dialog Box

The MIB Application Builder window contains a list of all existing MIB applications and the location of their selections in the NetView for AIX menu structure. You can select an existing application to modify or delete, or you can add a new application.

Adding a MIB Application: To add a MIB application, select the **Add...** button. The Add MIB Application dialog box contains four sections, each of which is described in the following paragraphs.

The top section contains text fields where you can enter an Application ID, or name, for your MIB application. This name should be the same as the name you specify in the menu path that enables users to access your application. Your MIB application will be available from the NetView for AIX main menu and from the object context menu.

The Application Title field enables you to specify the text that is displayed on the title bar in the application once it has been selected. You can select one of the following formats for your application's output:

- Form: Use only MIB objects that have a single instance per system, that is, those that occur only once.
- Table: No restrictions identified.
- Graph: Use only MIB object types Integer, Counter, and Gauge. Variables with these types can have more than one instance per system.

The Display Fields section contains data about the MIB objects you select to be monitored by this application. Select the **Add...** button to display the MIB Application

Builder/Add MIB Objects dialog box. Use this dialog box the same way you used the Browse MIB dialog box to find the MIB objects you want to add to your application.

The NetView for AIX Integration section is where you decide what menu path users must traverse to select your application. The default main menu selection is Monitor; you can choose selections under Monitor that are most appropriate for your application, or add selections if necessary. You can also change the selection rule, which directs the application to begin only if the selected objects meet the criteria found in the rule. For example, if the selection rule is:

```
(isSNMPSupported || isSNMPProxied)
```

the MIB application will run only if the objects selected on the map support SNMP directly or through a proxy agent. If you select an object that does not meet the selection criteria, the MIB application's menu selection is grayed.

In the bottom section, Help Text, you can enter some basic information about what your application does and how to use it. It is good practice to enter at least a minimum amount of information in case someone wants to know what your application does. A user can access Help for your application by selecting the **Help** button on the application's menu bar and selecting On Application.

Once you have completed all sections of the Add MIB Application dialog box, press OK to add the new MIB application to the list of available MIB applications. Press Close to exit from the MIB Application Builder.

Running MIB Applications through the Graphical Interface: To run the application, select an object or objects on the network map, select the Monitor option from the NetView for AIX main menu, and follow the menu path you entered for your application. If your application graphs MIB values, it runs in the Control Desk window until you stop it by selecting **File..Exit** from the application's menu bar. If your application produces a table or form, it runs in a separate window.

If a MIB application does not run, check the following conditions:

- Make sure that the MIB is loaded into the Loaded MIB Database. Sometimes, a MIB is loaded for the specific purpose of creating a MIB application, and then it is unloaded to conserve space in the database.
- Check that the MIB object ID you have selected is supported by the device you want to monitor.
- Make sure that the application was created to monitor the selected device and that the community name is properly set. Check the /usr/OV/conf/snmp.conf file to verify the community name.

Running MIB Applications Using the runapp Command: You can run an existing MIB application without using the MIB Application Builder. The **runapp** command executes the MIB application, using the application name and host name you specify, as shown in the following example:

```
runapp -a mibObjApp -h host1
```

For more information about the **runapp** command, refer to the man page.

Collecting Historical Performance Information

You can compile historical performance data about your network by using the NetView for AIX MIB Data Collector. This tool enables you to manipulate data in several ways, including:

- Collect MIB data from network nodes at regular intervals.
- Store MIB data in a file. If the NetView for AIX program is configured to work with a relational database, you can transfer collected data to a relational database and use the relational database tools to create reports. Refer to the *NetView for AIX Database Guide* for more information about transferring collected data to a relational database.
- Define thresholds for MIB data and generate events when the specified thresholds are exceeded. Setting MIB thresholds enables you to automatically monitor important network and system parameters to help you detect and isolate problems.

Using the MIB Data Collector

Before you configure your system to collect MIB data, you need to understand the definitions of the MIB objects and what they do. To look at descriptions for selected MIB objects, select **Tools..MIB Browser: SNMP** from the NetView for AIX main menu. Refer to the vendor documentation for information about their enterprise-specific MIBs.

You must also make sure that there is enough room to store data in the /usr/OV/databases/snmpCollect directory. It might be necessary to remove some files in this directory to make space available. You can set a **crontab** command to periodically remove files, or use the **snmpColDump** command to edit the directory files and delete selected lines. See the man pages for more information about using these commands.

Select **Tools..Data Collection & Thresholds: SNMP** from the NetView for AIX main menu. The MIB Data Collection dialog box is displayed, as shown in Figure 36 on page 196.

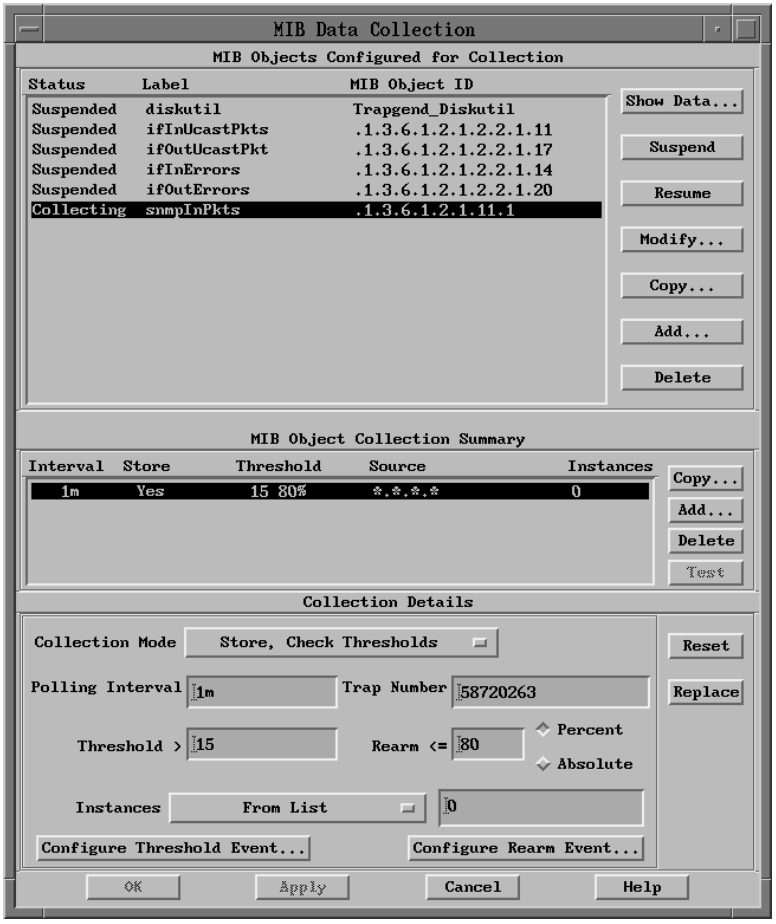


Figure 36. MIB Data Collection Dialog Box

The MIB Data Collection dialog box contains three sections:

- MIB Objects Configured for Collection

  Use this section to select the MIB objects or MIB expressions on which to collect data.

- MIB Object Collection Summary

  Use this section to identify the nodes from which to collect MIB data and to view the current collection defaults.

- Collection Details

  Use this section to identify details about the data collection such as how often to collect data and whether to send threshold events.

MIB Objects Configured for Collection: The MIB Objects Configured for Collection section of this dialog box contains the following fields:

Status — Indicates whether data collection is suspended or active. You can change the status by selecting either the Suspend or the **Resume** button.

Label — Displays the name of the file to which data will be stored.

MIB Object ID — Displays the object identifier of the MIB object configured for collection. If the value in this field does not begin with a dot (.), it is a MIB expression. This is the path through the MIB tree that ends in this particular leaf node, or object.

MIB Object Collection Summary: The MIB Object Collection Summary section of this dialog box contains the following fields:

Interval — Specifies how often data is collected from a source. The s, m, h, or w after the number indicates seconds, minutes, hours, or weeks.

Store — Indicates whether or not data is being stored.

Threshold — Displays two values: the first is the threshold, the second is the rearm. If the threshold and rearm values are not defined, the Threshold column displays `0.00 0.00`.

Source — Displays the name of the collection source. The source name can be an individual node or a set of nodes based on an IP address wildcard, for example, 15.212.*.*.

Instances — Displays the MIB instances on which data is to be collected. The instance is an internal counter on the system. For example, assume you have multiple disks on a node and the name of the MIB object is disks. The instance tells the data collector on which disk to collect data. An instance value that is an asterisk (.*) signifies that data is to be collected on all instances. A numerical instance value specifies a particular instance. An instance can also be a regular expression.

Collection Details: The Collection Details section of this dialog box contains the following fields:

Collection Mode — Determines the collection mode. Use one of the following four values:

- Exclude Collection
- Store, Check Thresholds
- Store, No Thresholds
- Don't Store, Check Thresholds

Polling Interval — Determines how often data is collected from a source. Enter a positive real number followed by an s, m, h, or w that indicates seconds, minutes, hours, or weeks. If you don't enter a letter following the number, the data collector uses the default of seconds. For example, `1.5h` indicates one and a half hours, and `30` indicates 30 seconds.

Trap Number	Specify an enterprise-specific trap number. The default trap is 58720263, the enterprise-specific trap that the MIB Data Collector sends when a threshold or a rearm value is exceeded. The MIB Data Collector sends the trap using the enterprise ID of the management station.
Threshold	Enter a threshold value that specifies when you want to be notified of traffic patterns that are outside the normal expectations. When the threshold value is passed, the specified threshold event is generated.
Rearm	Enter an appropriate rearm value to control the frequency of threshold events generated. When a MIB value drops below or is equal to the rearm value, a rearm event is generated. Another threshold event will not be generated until the rearm event occurs and the collected value again exceeds the threshold value after being rearmed.
Instances	Use the option list to select the type of instances you want to enter in the text field to the right of the option button. Specify the instance of the MIB object on which you want to collect data. If the object on which you want to collect data does not support multiple instances, the instance is zero. If you have multiple instances of a MIB object on a node, you must specify the instance on which you want to collect data. See the dialog box help for more information.

Example of Collecting MIB Data

Suppose you want to collect information about the number of inbound SNMP packets received by objects on a particular network. Follow these steps:

Step 1. **Tools..Data Collection & Thresholds: SNMP** from the NetView for AIX main menu.

Step 2. Select the snmpInPkts item from the default list of configured collections. The current status of snmpInPkts data collection is Suspended.

Note the information about this list item that is displayed in the MIB Object Collection Summary area of the dialog box. In this area, you can select the **Add...** button if you want to add other nodes from which to collect this data.

Step 3. Select the list item in the MIB Object Collection Summary area of the MIB Data Collection dialog box. The bottom area of the dialog box, Collection Details, becomes active and displays the values from the Collection Summary area in fields that you can change.

The following values apply to this particular collection:

- The polling interval is 1 hour.
- The data collected is to be stored and checked against a threshold value.
- When the collection frequency exceeds 15 occurrences per hour, a threshold event will be generated, sending trap number 58720263 to the manager.

- Once the trap has been sent, the rearm value, 70%, controls the frequency with which subsequent traps signifying a threshold event will be sent.

Step 4. If this information suits your needs, you can select the **Resume** button and then the **Apply** button to apply the change in status.

Notes:

a. Although the Status field for snmpInPkts changes to Collecting after you select the **Resume** button, the change does not take effect until you select the **Apply** button at the bottom of the dialog box.

b. Data collection is restarted each time you select the **Apply** button, which means that polling is interrupted. For example, suppose that you have set polling intervals on a MIB object to 1 hour. Selecting the **Apply** button causes the data collector to begin polling, even if the last poll took place only 5 minutes ago. Selecting the **Apply** button several times will increase network traffic and store more data than you might have intended.

Step 5. To see the collected data, select the **Show Data...** button in the MIB Objects Configured for Collection section of the MIB Data Collection dialog box. Collected data will be displayed after the first polling interval you specified has passed. In this example, no data will be displayed until one hour has passed from the time you clicked on the **Apply** button to resume data collection for snmpInPkts.

The data is displayed as a table that lists the polling interval, time of collection, the source node from which the data was collected, and the value of snmpInPkts for that node. If you want to see a graph of the collected data, select the **Graph** button at the bottom of the MIB Data Collection / Show Data dialog box.

Step 6. You can exit from the Tools..Data Collection & Thresholds: SNMP operation and view the collected data at a later time by selecting **Tools..Graph Collected Data: SNMP** from the NetView for AIX main menu. Again, you must wait until after the specified polling interval has passed before there will be any collected data to view.

If no data is being collected, make sure that the snmpCollect daemon is running on the manager. This daemon stops running when file system space is not available. If you have root authority, you can use SMIT to restart the daemon. Then you can restart the data collection.

Refer to the man page for more information about the snmpCollect daemon.

Graphing MIB Variable Expressions

You can graph the result of expressions applied to MIB variables. Expressions for manipulating MIB variables are stored in the /usr/OV/conf/mibExpr.conf file. These expressions are in postfix format. You might want to try using the example expressions before you create others.

Steps: To create and graph the results of an expression, take the following steps:

Step 1. Use one of the example expressions or add an entry to the /usr/OV/conf/mibExpr.conf file.

Step 2. Select an object or objects on a submap.

Step 3. Select **Tools..Data Collection & Thresholds: SNMP** from the NetView for AIX main menu.

Step 4. Select the **Add...** button in the MIB Data Collection dialog box.

Step 5. Select **Expression** in the MIB Data Collection / MIB Object Selection dialog box.

Step 6. Select an expression from the list that is displayed in the Expression ID text field and select OK. The MIB Data Collection / Add Collection dialog box is displayed.

Step 7. Select the type of instances you will enter in the Instances text field to the right of the Instances option button. Or, enter `.*` to collect all instances of the selected MIB variables.

Step 8. Select the **Add From Map** button to display the selected object or objects in the List of Collection Sources field.

Step 9. To change the collection mode, select the **Collection Mode** list button. There are four selections from which to choose:

- Exclude Collection
- Store, Check Thresholds
- Store, No Thresholds
- Don't Store, Check Thresholds

You can also change the threshold and rearm values if necessary. Select the **OK** button to close the dialog box.

Step 10. The Status field in the MIB Objects Configured for Collection area of the MIB Data Collection dialog box now shows Collecting for this MIB object. However, you must select the **Apply** button to actually start the data collection.

Step 11. Once data has been collected, you can display it by selecting the **Show Data..** button in the MIB Data Collection dialog box. The data is displayed in table format, but you can convert it to a graph by selecting the **Graph** button at the bottom of the MIB Data Collection / Show Data dialog box. You can also graph the collected data at a later time by selecting **Tools..Graph Collected Data: SNMP** from the NetView for AIX main menu.

Using the setthresh Command

You can set a threshold for MIB data collection without using the Tools..MIB Data Collection operation. The **setthresh** command sets up data collection configurations and stores the values in the /usr/OV/conf/snmpCol.conf file. If any entry already exists with matching MIB object ID and source name, that entry is updated with the new information.

Example of Collecting Thresholds: You can collect data or monitor thresholds only on numeric MIB variables, that is, those that are defined as type Counter, Gauge, or Integer. The following example shows how the **setthresh** command is used to collect the same data that was collected in the example above:

```
setthresh -o snmp.snmpInPkts -s R
          -n *.*.*.*
          -c W -m s -p 1h
          -v 10 -r 70 -t %
          -i ALL -T 58720263
```

For more information about the **setthresh** command, refer to the man page.

Monitoring File System and Paging Space

Collecting network and performance data can quickly deplete key system resources. If file system or paging space become full, some processes might stop. However, it's not always convenient to continually monitor the system for this problem.

When you want to monitor file system or paging space, you can use the Monitor..Local Filesystem and Paging Space option to receive a trap or a message that informs you when a threshold condition has reached its limit. This option monitors the root file system and the paging space on the local manager system where NetView for AIX is installed (usually /usr/OV). If /usr/OV/databases and /usr/OV/log are defined as separate file systems, the program monitors them as well.

The file system and paging space monitoring process uses the values set in the Monitor File System & Paging Space dialog box.

The Monitor File System & Paging Space Dialog Box

When you select **Monitor..Local Filesystem and Paging Space** from the NetView for AIX main menu, the Monitor File System & Paging Space dialog box is displayed as shown in Figure 37 on page 202.

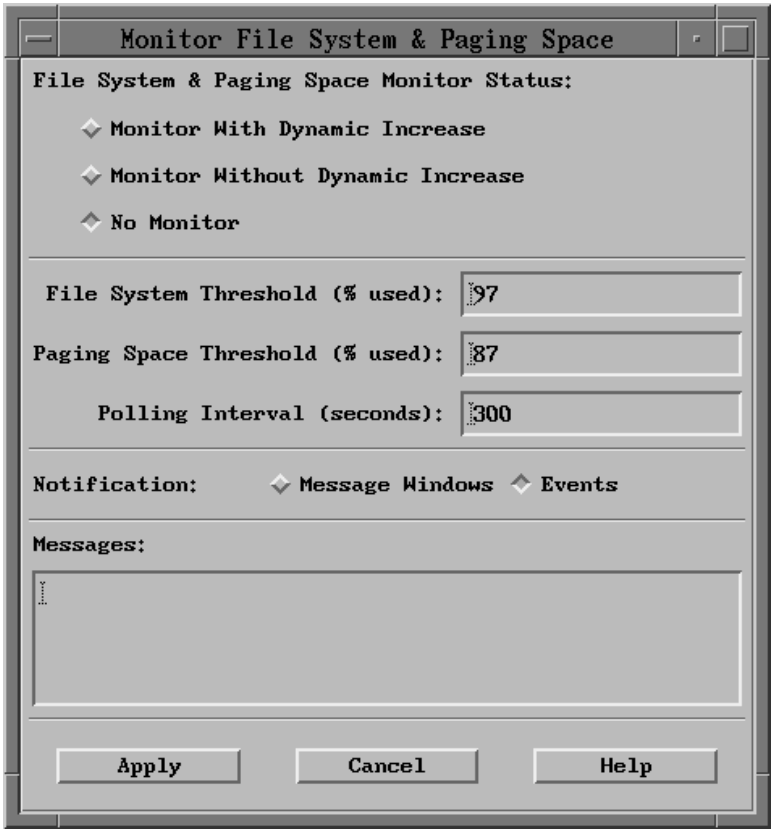


Figure 37. Monitor File System & Paging Space Dialog Box

You can start file system monitoring (with or without dynamic increase of file system space) and accept or change the default values for file system and paging space thresholds, the polling interval, and the notification method. The Monitor File System & Paging Space dialog box contains the following settings:

- File System & Paging Space Monitor Status
  - Select the **Monitor with Dynamic Increase** button if you want to start file system and paging space monitoring and if you want file system size or paging space to be increased automatically when the threshold is reached.

    File system size is increased by 4 MB, and paging space is increased as follows:

    - If only one page space is defined and there are free partitions on the hard disk, the paging space is increased by 12 MB.
    - If more than one page space is defined and the page space with the smallest percentage of use has reached the threshold, the page space defined on the hard disk with the most free partitions is increased by 12 MB.

- If more than one page space is defined on a single hard disk, then the page space with the smallest percentage of use is increased by 12 MB.

  If no free partitions exist, users are notified by the notification method you specify (a message window or a trap)

The monitoring process also logs a message to the /usr/OV/log/shpmon.log file, indicating when the process was started, stopped, and the amount of increase that was made to the file system space or paging space.

– Select the **Monitor Without Dynamic Increase button** to start file system and paging space monitoring and to notify users when file system or paging space needs to be increased.

- File System Threshold

  The default threshold for file system space is **97** percent. You can use this value, or you can type a different value in this field. Users are notified when the threshold is reached.

- Paging Space Threshold

  The default threshold for paging space is **87** percent. You can use this value, or you can type a different value in this field. Users are notified when the threshold is reached.

- Polling Interval

  The default polling interval is **300** seconds. You can use this value, or you can type a different value in this field.

- Notification

  You can notify users with an audible message box or by generating a trap. The default is to generate a trap. You can configure the trap to be displayed and logged using the Options..Event Configuration..Trap Customization: SNMP operation.

Starting and Stopping the Monitoring Process

To start or stop the file system and paging space monitoring process, follow these steps:

Step 1. Select **Monitor..Local Filesystem and Paging Space** from the NetView for AIX main menu.

The Monitor File System & Paging Space dialog box is displayed as shown in Figure 37 on page 202.

Step 2. Make the appropriate changes to the dialog box. See “The Monitor File System & Paging Space Dialog Box” on page 201 if you need more information about the dialog box settings.

Step 3. Select the **Apply** button to accept the changes and close the dialog box.

Note: If you stop the monitoring process when the process is inactive (depending on the polling interval), it seems as though the process is still running. Because the termination signal has not yet been received if you attempt to start the moni-

toring process again, a message is displayed telling you that the process is already running. When the monitoring process becomes active, the monitoring process receives the termination signal and stops.

Starting from the Command Line

You can start the file system and paging space monitoring process from the command line and specify what options to use. Use the following command to start the monitioring process:

```
shpmon [-m 1|0] [-e] [-t update_time][-Q]
```

Where:

-m 1|0 Starts with or without dynamic increase. The value 1 starts monitoring with dynamic increase. You must be a root user to use this option. A 0 (zero) value starts monitoring without dynamic increase.

-e Executes the program one time and exits. The monitoring process checks the thresholds set in the Monitor File System & Paging Space dialog box and responds depending on what other flags are used. See “The Monitor File System & Paging Space Dialog Box” on page 201 for more information.

-t Specifies a polling time interval in seconds. If this option is not specified, the value set in the Monitor File System & Paging Space dialog box is used. See “The Monitor File System & Paging Space Dialog Box” on page 201 for more detailed information.

-Q Suppresses all messages sent to the user when the threshold value has reached its limit. Messages about changes to the file system and paging space sizes are logged in the shpmon.log file. Events are generated when threshold values are exceeded and cannot be increased.

Monitoring Specific Events

To configure the Local Filesystem and Paging Space option to execute only when a specific event occurs, use the Event Configuration dialog box. When a threshold event occurs for the configured MIB object ID, the program executes, takes the appropriate actions, and exits.

Follow these steps to configure system events. You must be a root user.

Step 1. Select **Tools..Data Collection & Thresholds: SNMP** from the menu menu. On the MIB Data Collection dialog box, configure the threshold value and trap number for the disk utilization MIB object. See “Using the MIB Data Collector” on page 195 for those steps.

Step 2. Select **OK**.

Step 3. Select **Event Configuration..Trap Customization: SNMP** from the Options pull-down menu. On the Event Configuration dialog box, select the same trap number you configured on the MIB Collection dialog box. The trap

numbers are listed in the Event Identification section under Event. Use the Help option if you are not familiar with the Event Configuration dialog box.

Step 4. Select **Add**. The Add Event dialog box is displayed.

Step 5. Enter the following command in the Command for Automatic Action field:

```
/usr/OV/bin/shpmon -m 1 -e -Q
```

Step 6. Select **OK** to close the Add Event dialog box.

Step 7. Select **OK** to close the Event Configuration dialog box.

This command executes the shpmon process only one time when the specified event occurs. It checks the file system and page space, increases them if necessary, and exits.

Using the NetView for AIX Graph Applications

Graph applications provide a convenient way to display performance information. This section describes how to use and customize graph applications.

Starting a Graph Application

To start a graph application, select an object or objects on a submap, then select the graph application whose results you want to view. All graph applications can be started from either the NetView for AIX main menu or the object context menu. The graphs are displayed in a control desk window.

MIB Graph Applications

The following MIB graph applications can be started from an object context or main menu:

- Monitor..System Activity..CPU Performance
- Monitor..System Activity..Disk Space
- Monitor..Network Activity..Interface: Traffic
- Monitor..Network Activity..Ethernet..Traffic
- Monitor..Network Activity..Ethernet..Errors
- Monitor..Network Activity..TCP: Connections
- Monitor..Network Activity..SNMP..Traffic
- Monitor..Network Activity..SNMP..Operations
- Monitor..Network Activity..SNMP..Errors
- Monitor..Network Activity..SNMP.. Authentication Failures

If you start one of these applications from the Tools window, you can place it in a control desk or drop it anywhere else on the desktop to make it a stand-alone application.

If you start one of these applications from an object context menu, it is displayed in the control desk. Starting one of these applications from an object context menu means it applies only to the selected object. If you want to collect and graph data for more than

one network object, select the objects on the submap, then select the appropriate application from either the NetView for AIX main menu or the Tools window.

Saving Performance Data

To save the performance data that is displayed in the graph application, select **File..Save As..** from the menu bar in the application window. You can save the data to the default directory and file name, /tmp/xnmgraph.data, or specify a different directory and file name. Select the **Apply** button to save the data. Look in the Messages area for any messages generated as a result of the Apply operation. For example, if the save is successful, the message tells you how many data points were saved for each monitored MIB object.

The Save As.. operation dumps an ASCII file containing the data to the directory and file name you specify. The ASCII file displays the collected data in table form for each monitored MIB object.

To re-create the graph from the contents of the ASCII file, enter the following command in an aixterm window:

```
cat /u/<userid>/<directory>/<filename.ext> | /usr/OV/bin/xnmgraph
-title "Monitor: SNMP Traffic"
-units "Packets / Sec"
-helpFile ovip/OVW/Functions/snmpTraffic
-mib "-1:Input Packets:::::::602route1.raleigh.ibm.com,-2:Output
Packets:::::::602route1.raleigh.ibm.com,"
```

Printing Graphed Data

To look at some of the graph information in printed format, print the contents of the graph window. Select the Print Tool icon in the Tools window and drag it to another area on the desktop to open a Print Tool window. Follow these steps to capture and print the contents of the graph window:

Step 1. Rearrange the windows on your desktop so that the window containing the graph display does not overlap with any others.

Step 2. In the Print Tool window, select the **Capture** button.

Step 3. Move the pointer, which has changed shape from an arrow to a hand, to the graph window you want to capture and click mouse button 1.

Step 4. In the View Identification dialog box that is displayed, enter a file name for the captured information. Select the **OK** button to close the View Identification dialog box.

Step 5. The file name you specified is displayed in the Selected field of the Print Tool window. You can either print the captured information immediately or save it to print later.

Before you print, it might be helpful to save the captured information in a directory in case you need to print it again.

Step 6. To print the captured information immediately, select the name of the file and specify the name of a printer and the number of copies, then select the **Print** button. The Print Tool window closes.

Step 7. To save the captured information, select the name of the file and then select **File..Save** from the menu bar in the Print Tool window. In the File Save dialog box, enter the directory path where you want to store the file. You do not need to enter a file extension because the Print Tool automatically adds `.ps` to the file name if you save the captured image in a directory. Select the **OK** button.

Step 8. To exit the Print Tool, select **File..Exit** from the Print Tool menu bar.

Note that the Print Tool captures and prints only the information currently contained in the graph window. It does not print the entire contents of the graph. If you need to print the entire graph, you can page forward and backward and use the Print Tool to print each page of graphed information. To page forward or backward, click anywhere on the graph itself and select Page Forward or Page Backward from the context menu that is displayed.

Rate of Change	The default value, which shows the new counter value as a time-averaged value since the last query for the MIB object.
Actual Sampled Value	Shows the actual value returned from the MIB Counter variable.
Delta Value	Shows the actual change in the MIB variable since the last query for the MIB object. This value is not time-averaged.

Example

For example, suppose you are graphing a MIB variable with the following statistics:

```
Value of MIB variable at time 0  --> 100
Value of MIB variable at time 10 --> 300
```

The value from `time 10` would be graphed in the following ways:

Rate of Change	20 (derived from (300–100)/10)
Actual Sampled Value	300
Delta Value	200 (derived from (300–100))

Adding a Line

An application sometimes has more lines to graph than the maximum number that are specified for the graph when the application is created. You can use the **View..Add Line** operation to temporarily increase the number of lines the graph can display. This selection will be grayed out if the number of lines to be graphed does not exceed the maximum number of lines that can be graphed, as defined in the application's app-defaults file.

Using the Context Menu

The context menu in a graph application enables you to zoom in and out so you can look at the graph from different perspectives. You can also use the context menu to page forward or backward through the collected data, or to display the beginning of the data, the end of the data, or all of the data.

Generating Performance Reports

The NetView for AIX program provides sample shell scripts that use real-time and historical information to generate performance reports. In addition to using these shell scripts, you can create others that collect and display the performance information you most need to see. These reports are stored in the /usr/OV/reports/C directory.

Select **Monitor..Reports: Site Provided** from the NetView for AIX main menu. The sample shell scripts provided with the NetView for AIX program are displayed in the Run Report File window.

Figure 38 shows the reports shipped with the NetView for AIX program.

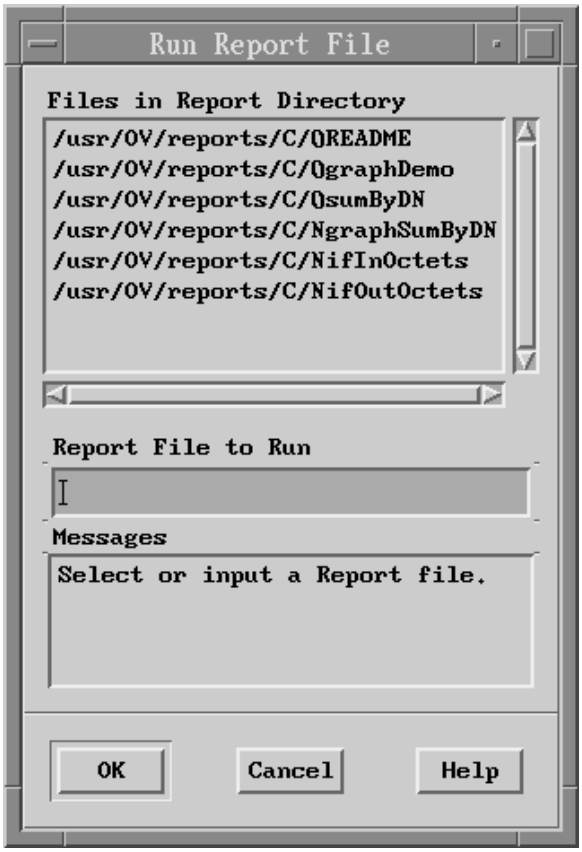


Figure 38. Sample Shell Scripts in the NetView for AIX Report Directory

You can create your own reports to gather real-time or historical information about MIB objects. Your reports can be stored in the /usr/OV/reports/C directory, or in another directory of your choice. When you want to run a report that is stored in another directory, enter the full path name of the report in the Report File to Run field of the Run Report File dialog box.

Contents of the Report Directory

This section describes the files and shell scripts that are provided in the NetView for AIX reports directory.

- The 0README file provides a description of each report in the /usr/OV/reports/C directory, including the type of object each report requires as a selection. It is good practice to update the 0README file each time you create a report and add it to the report directory.
- The 0graphDemo shell script provides an example of using the NetView for AIX graph facility, called xnmgraph, to graph information that is collected in a standard AIX file.
- The 0sumByDN shell script uses data from the SNMP Data Collector to provide an approximate estimate of which nodes have the most total traffic. This shell script must be run with data of type Counter.
- The NgraphSumByDN shell script uses data collected by the SNMP Data Collector to estimate which nodes have the most total traffic by time of day. Only nodes that are currently selected will be graphed.
- The NifInOctets shell script shows how to use the NetView for AIX graph facility, xnmgraph, to graph a specific MIB object for selected nodes.
- The NifOutOctets shell script shows how to use the NetView for AIX graph facility, xnmgraph, to graph a specific MIB object for selected nodes. The values must have been collected by the SNMP Data Collector.

Writing Reports

You can write reports that send their output directly to the terminal so you can see the results immediately. Getting immediate feedback can help you with monitoring and troubleshooting. You can also write reports that store their results in a file or in a database, where they can be used to chart performance trends over a period of time.

Some reports are run only if an object is selected from the map before the report is selected. Other reports require that more than one object be selected, or that no objects be selected. You can tell which reports require selected objects, and how many selected objects, by looking at the first character of the report name.

Table 15. How Many Selected Objects a Report Requires

First Character of Report Name	Number of Selected Objects
0	None
1	One
N	One or more

You might want to follow this convention when you create reports.

Chapter 8. Using the Agent Policy Manager

If you have the Systems Monitor Version 2 Mid-Level Manager or System Information Agent installed in your network, read this chapter to understand how to configure and use the Agent Policy Manager to more easily set up and view information about thresholds and file monitoring in a network. To use all the functions of APM, you must have at least one Mid-Level Manager (for thresholding) and one or more System Information Agents (for file monitoring) installed somewhere in your network.

This chapter is organized to accommodate both users who are familiar with Systems Monitor and those who are new to it.

"What the Agent Policy Manager Can Do For You" gives an overview of the Agent Policy Manager and its relationship to Systems Monitor, the Collection Facility, and NetView for AIX.

"Configuring and Starting APM" on page 213 explains how to configure and start the Agent Policy Manager.

"An Example of Defining and Distributing A Definition" on page 215 provides a real-life scenario of how a user would define a file monitor condition.

"Using the APM Interface to Define and Distribute Definitions" on page 218 describes how to use the Agent Policy Manager interface to define and distribute thresholds and file monitor conditions.

"Diagnosing Problems Using the Problem Determination Assistance Facility" on page 232 describes how to use the PDA facility under APM.

"Agent Policy Manager Reference" on page 233 has detailed information for those readers who are interested in how the Agent Policy Manager defines itself and its domains in the network, how community names are configured, and other such reference information.

What the Agent Policy Manager Can Do For You

If you use Systems Monitor V2 in your network, you know that it is a powerful application. Through the Mid-Level Manager (or MLM), you can offload some systems management from NetView for AIX with the MLM's thresholding capability. An MLM can identify a developing problem and alert you to it or take corrective action before your users ever become aware of it. The System Information Agent, or SIA, has an extensible MIB that you can use to monitor not just the machine, but processes and applications running on the machine. Through the SIA File Monitor table, you can monitor a file for a wide variety of conditions, such as for a specific string of text or even for a change to the group or user assigned to a file, and execute a command or shell script when the specified condition occurs.

Systems Monitor is powerful, but it can require that you gain many new skills, especially if you are new to SNMP and the concepts of MIBs, community names, and SETs and GETs. The APM interface, running in conjunction with Systems Monitor and the Collection Facility, allows you to accomplish these same tasks more easily by:

- Simplifying the task of defining threshold or file monitoring conditions by eliminating the need to define trap destinations.
- Providing the ability to distribute a threshold or file monitor configuration to groups of nodes as a single operation.
- Automatically creating submaps and an icon on your NetView for AIX root map for MLMs and their managed nodes
- Automatically creating icons on your root map that represent active threshold and file monitoring settings
- Providing a way to filter file monitor traps so that only the ones you care about are forwarded to the NetView for AIX Control Desk.

This management of agents in the network is accomplished by setting up *policies*. The policies you define are made up of two pieces:

- Defining rules about **who** will be acted on (such as all routers, all machines in a building, or all devices in a certain subnet)
- Defining rules about **what** action will be taken (such as threshold on a MIB-II variable, monitor a log for an error message)

APM simplifies the task of distributing changes to your network. Rather than NFS mounting Systems Monitor configuration files, or editing configurations on every MLM or SIA machine in your network, you use the Collection Facility to set up collections to include all the nodes that will get a new threshold or file monitoring definition. After you have a collection defined, it is a simple matter of clicking on the **Distribute** button to update all the nodes in your network. The Collection Facility maintains a list of objects that fit collection rules. It updates the list if changes in the network topology result in the addition of nodes that fit a collection rule (or deletion of nodes already in a collection). Agent Policy Manager will act on these changes by configuring new nodes that fit the collection rule or removing definitions from nodes that no longer fit collection rules. This collection maintenance takes place automatically.

See “Defining and Managing Collections of Objects” on page 81 for more information.

APM Collection Icons That You Get Automatically

APM and the Collection Facility create several new icons on the root map:

- **Collections** shows all collections that have been defined. It is added to the root map by the Collection Facility. Double-clicking on this icon displays a dialog box showing all of the defined collections. Initially, these collections are unmanaged.
- **MLM Managers** is a collection of all the MLMs in your network, with all the IP nodes in each MLM's domain. Double-clicking on this icon displays a submap of all MLMs in the network. Double-clicking on an MLM displays a star configuration of the MLM and the nodes in its domain.

  Double-clicking on a particular node shows a submap of the node and details about the node itself.

- **APM Monitors** is displayed after you define a threshold or file monitor condition. This icon indicates that thresholding and file monitoring definitions are active.

  Double-clicking on this icon displays a dialog box showing the threshold and file monitor definitions. Double-clicking on one of these definitions shows the nodes in the collection against which the definition is set.

  Double-clicking on an individual node displays details about the node itself, including IP status and icons for the individual threshold or file monitor definitions. The threshold and file monitor definitions are executable icons. Double-click on this icon to access the PDA menu. The PDA menu is described in "Diagnosing Problems Using the Problem Determination Assistance Facility" on page 232.

Propagation of Threshold or File Monitor Status to Map Icons

The color of each icon reflects the status of the threshold. On the NetView for AIX root map, if a threshold has been exceeded, the APM Monitors icon turns yellow. Similarly, if a file monitor condition has been detected, the icon turns yellow. Threshold and file monitor statuses are propagated upwards to the Collection icon on the root IP map. The status of the session between a managed node and its managing MLM is also propagated upward. If there is more than one threshold or file monitor condition defined, the aggregate status of the definitions is shown on the IP root map.

At the node level view from the APM icon, the icon for the actual threshold or file monitor condition that was reached is red; other icons do not change color. Similarly, if the session between the MLM and the managed node has gone down, the SM Map icon is red.

The MLM threshold function has the ability to rearm itself (that is, to reset the status of the threshold) when the value of the MIB variable being monitored reaches a rearm value you have defined. The color of the Agent Policy Manager Monitors icon will be reset to green if this was the only threshold that had been reached.

File monitor functions do not have a rearm definition. Thus, if the condition being monitored for is found and the Agent Policy Manager Monitors icon changes color, you must reset the color manually through the PDA menu. The PDA menu is described in "Diagnosing Problems Using the Problem Determination Assistance Facility" on page 232.

Configuring and Starting APM

As shipped, the APM function is not automatically configured to run. To use the APM, you must configure and start the daemon through SMIT.

Configuring the APM daemon Through SMIT

1. From the SMIT main panel, select the following menu options:

   Communications Applications and Services →
   NetView for AIX →
   Configure →
   Set Options for Daemons →
   Set options for Agent Policy Manager daemon

Through this option, the APM application is registered and defaults for logging and tracing.

2. Change default log and trace files if desired.
3. Optionally, change the elapsed time between daemon attempts. This value determines how often APM will attempt to distribute threshold and file monitoring definitions that it could not distribute previously (for example, if a node was down). By default, Agent Policy Manager will retry all failed distributions for all nodes once every 60 minutes.
4. Optionally, change the number of threshold events stored in the history file. These events are used as plot points by the xnmgrapher application. You can graph a threshold's history through the PDA Application.
5. Select **Do**.

After you have registered and started the daemon, it will be started automatically when you start NetView for AIX.

Starting the APM daemon from the Command Line

After the APM daemon is configured, you can start it from the command line. The daemon is called C5d. You need to have root authority to start the daemon. Enter the command **/usr/OV/bin/ovstart C5d** on the command line.

The following options can be specified when you start the C5d daemon:

-T Toggle C5d tracing.

-t *tracefile*
Turns on tracing when the C5d daemon is started. Trace information is saved in the file specified.

-L Toggles C5d logging.

-l *logfile*
Log file is saved in the file specified.

-n Turns off logging at C5d daemon startup.

-g Specifies the number of threshold events to be saved.

-r Specifies the retry interval.

See the C5d man page for more information.

Starting the APM Configuration Interface

The APM Interface is integrated into the NetView for AIX map interface. You can start it from two places:

- Select **Tools...APM Configuration** from the NetView for AIX menu bar.
- Double click on the APM icon on the Tools Window.

From the APM main dialog, you can look at a list of current log and threshold definitions. To switch from threshold to log definitions, use the toggle button. You can also add, delete, or modify a file monitor or threshold definition, and distribute the definition to other nodes in the network.

Note that the first time you start the APM EUI, it might take some time to appear, because it must set status on all nodes in the associated collections Agent Policy Manager uses and create submaps for the collections. After this initial synchronizing has been done, you will not see a delay in the EUI startup.

You can use the Administer..Start Application..APMmap menu option to start the Agent Policy Manager map application without starting Agent Policy Manager configuration and without have to close and restart the graphical interface. You might find this useful if the map application ends abnormally or if you start the Agent Policy Manager daemon after you have started the NetView for AIX graphical interface.

An Example of Defining and Distributing A Definition

To help you understand how APM interacts with the Collection Facility, NetView for AIX, and Systems Monitor, read this step-by-step scenario of how an administrator would define a collection and set up a file monitoring condition.

1. A network administrator at a university has been having problems with someone logging into the AIX fileservers in the network and tampering with files to which only root has access. On the assumption that the troublemaker is guessing at root passwords, and thus making several faulty login attempts each time a machine is broken into, the administrator decides to monitor the /etc/security/failedlogin file for failed attempts. He will use the file monitoring capability provided by the Systems Monitor System Information Agent, which is installed on each of the fileservers. Since the /etc/security/failedlogin file is in binary, the administrator will use the **fwtemp** accounting command provided on AIX to translate the file to ASCII before file monitoring begins.
2. First, the administrator defines a collection called `fileservers` that includes the important nodes. He starts the Collection Editor from the NetView for AIX Tools menu and selects Add to add a new collection. On the Add Collection dialog, he defines a collection called `fileservers`, using the Node List definition type to specify the IP addresses of the AIX fileservers. He selects **OK** to add the collection.
3. Next, he selects **Tools...Agent Policy Manager EUI** from the NetView for AIX menu bar. He selects the **Add/Copy** button to add a file monitoring condition.
4. On the File Monitor dialog box, the administrator defines the following entries:

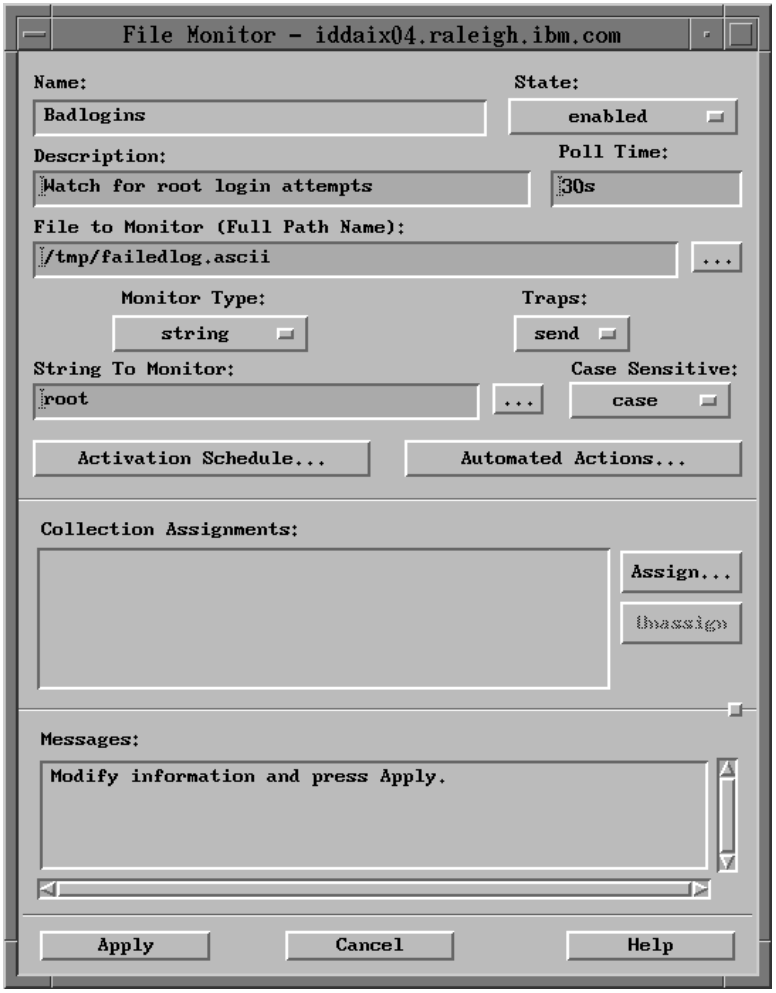


Figure 39. An Example File Monitoring Condition

The file monitor entry Badlogins will check the /tmp/failedlog.ascii file for the string `root` every five minutes.

The administrator clicks on Automated Actions. He defines the following entries:

Figure 40. An Example File Monitoring Condition - Automated Actions

The command `/usr/sbin/acct/fwtmp < /etc/security/failedlogin > /tmp/failedlog.ascii` uses the fwtmp utility to convert the binary file to ascii and put the translated information into a temporary file called failedlog.ascii.

After this command is executed, file monitoring begins. Every five minutes, the file is tested for the string `root`. If the string is found, a special file monitor trap is sent to NetView for AIX.

By default, the SIA would mark the file at the location where it stopped monitoring and would resume monitoring at that point in the file. In this particular installation, that default has been changed to reset monitoring to the beginning of the file between polling intervals. To prevent the SIA from generating another trap in the next polling interval for the string that was already found in the file, the administrator specifies the command `grep -v root /tmp/failedlog.ascii | /usr/sbin/acct/fwtmp -ic > /etc/security/failedlogin` as the command to be executed after the string is found. This command takes all contents of the failedlog.ascii file except the failed root login and writes the contents back into the failedlogin file. During the next polling cycle, when the binary file is again converted into ascii, it will not have the records that caused the previous file monitor trap to be sent.

5. Next, the administrator needs to specify collections to which this file monitoring definition will be applied. On the Agent Policy Manager main dialog, he clicks on the **Assign** button. On the Collection Assignments dialog that is displayed, he selects `Fileservers` and then clicks on **Assign**. Fileservers moves to the Assigned Collections list. He clicks on **OK** to make the assignment. (He could have also started the Collection Editor and defined the collection from this dialog if it was not already defined.)
6. On the File Monitor dialog box, the administrator clicks on **Apply** to apply the setting. A message is displayed in the Messages area that indicates the definition was saved successfully.
7. Now the administrator needs to distribute the definition to the machines that make up the fileservers collection. From the APM main dialog, he selects the badlogins definition and clicks on **Distribute**.
8. On the Distribute Definitions dialog that is displayed, he clicks on **Start** to distribute the collection. Messages indicate when the SET to each node is successful. (If one or more of the SETs had failed, a message indicating that the definition was only partially displayed would have been distributed.)
9. The administrator goes back to the NetView for AIX Control Desk and uses Create -> Dynamic Workspace to start a dynamic workspace. On the dialog that is displayed, he selects **Filter Activation** in the Filter section of the dialog. On the dialog box that is displayed, he selects File List to get a list of filters. From the list, he selects `C5filters` and clicks on **OK**. In the Available Filters in File section of the dialog box, he selects `C5Console` and Activate. Then he selects **Close**. He selects **OK** to open the workspace.
10. Later that afternoon, the Collections icon on the root map turns yellow. The administrator quickly checks the dynamic workspace he set up earlier and sees that file

monitor specific trap 21 (String Found) arrived. He has caught the perpetrator in the process of breaking into one of the fileservers.

11. Some time later (after locating the guilty party in the university computing center), the administrator double-clicks down through the APM Monitors submaps to the node-level view map for the affected node. He double-clicks on the executable icon to display the PDA menu. He resets the status on the node so that the color returns to green.

Using the APM Interface to Define and Distribute Definitions

This section explains how to use the APM graphical interface to define and distribute threshold or file monitor conditions. The process for adding a new definition is made up of two steps:

- Creating the new definition
- Distributing the definition to the nodes in the collection.

  Distributing the definition is described in “Distributing A Definition to Remote Nodes” on page 219.

You can define thresholds or file monitor conditions using the APM EUI.

This section does not supply details about what information to put in the fields on the Threshold and File Monitor configuration windows. If you are new to Systems Monitor and you need to understand what information should go in these fields, refer to the online help from these dialogs, or see the *Systems Monitor User's Guide*.

Creating a New Threshold or File Monitor Definition

You can create both thresholds and file monitor definitions from Agent Policy Manager main dialog. By default, File Monitor is selected as the type of definition that is to be created. Use the selection button at the top of the dialog to change the selection to Threshold/Data Collection.

To create a new threshold or file monitor definition, select **Add/Copy** from the APM main dialog. The dialog for defining a new threshold or file monitor definition is displayed. Fill in the fields, specify the collection the threshold or file monitor condition is for, and select **Apply**. (Note that if you do not have the collection defined before bringing up the Agent Policy Manager interface, you can start the Collection Editor from the Agent Policy Manager interface and define the collection.) You will see a message indicating that the definition was saved and that you can distribute it from the Agent Policy Manager main dialog.

Using an Existing Definition to Create a New Definition

If you want to base a new definition on one that you have already defined, select the definition you want to copy, and select **Add/Copy** from the APM EUI main dialog. Note that the new definition you are creating must have a unique name across all APM definitions. In other words, you cannot have two thresholds or two file monitor conditions

called Test, nor could you have one threshold called Test and one file monitor condition called Test.

The first time you start the Agent Policy Manager EUI, you will see two default definitions. One is a log monitoring definition, and the other is a threshold definition. It is suggested that when you create new definitions, that you select one of these defaults and use the **Add/Copy** button to copy the information for a new default. If you create a definition from scratch, and you leave some fields blank on the Threshold or File Monitor dialog, you might get unexpected results.

Modifying an Existing Definition

After a definition has been defined or distributed, you can modify the definition in two ways:

- Change the **who** by adding or deleting target collections from the definition.

  When you use the Collection Editor to change the collections used by an Agent Policy Manager file monitor or threshold definition, the change to the collection set is distributed to the nodes in the collection automatically. You do not need to redistribute the definition to the nodes in the collection to implement the changes to the collection.

- Change the **what** by modifying the threshold or file monitoring rule.

  After you modify a threshold or file monitor condition, you must redistribute the definition to the target nodes. The modified definition appears on the APM Distribute dialog with a status indicating that the redistribution of the modified definition is pending. You can choose to distribute the modified definition, undo the changes to the definition, or delete the definition completely.

Distributing A Definition to Remote Nodes

After you have successfully applied the threshold or file monitor condition, you can distribute it to the nodes specified in the collection assigned to the definition. From the APM main dialog, select the definition you want to distribute by clicking on it, and then select **Distribute**. The dialog that is displayed lists the name of the threshold or file monitor definition and the collections to which it will be distributed. Select **Start** to distribute the definition. Messages about the success or failure of the distribution appear in the Messages area on the Distribute dialog box.“Distribution Status Indicators” on page 238 explains these status messages.

For thresholds, the distribution process consists of SNMP SET commands on the specified MIB object on the MLMs assigned to nodes in the collections. Thus, you must have the community names on the node where you are running APM and the target MLMs set up so that the APM node is allowed to do SNMP SETs on the target nodes.

Note that you do **not** have to redistribute a threshold or file monitor definition if nodes have been added or deleted to target collections or if you have modified the collection rule in some way. The information is automatically redistributed by APM.

Successful Distribution

If all nodes in the collection are successfully modified, the status in the APM main dialog for the collection is modified to indicate that the definition was distributed.

Distribution Failures

If any or all of the distributions fail, the definition is marked as partially distributed on the APM main dialog. Failures can occur for several reasons:

- An incorrect community name (causes an authentication error)
- A network failure
- An SNMP agent or application not responding
- The target MIB is not loaded
- The distribution attempt times out

If you receive a message that targets are not defined, it indicates one of two conditions:

- On a file monitor definition, the target collection is empty
- On a threshold definition, the Agent Policy Manager either could not find an MLM in the network where the nodes in the collection are, or it could not find any MLMs at all.

You can continue to attempt to redistribute definitions by hand, or you can let APM continue to try to redistribute. APM will continue to periodically retry the distribution in the following ways:

1. Periodically depending on the interval set when the C5d daemon was started. (This option can be set using the –r option.) All failed distributions are tried at the same time. The default interval is to retry distribution once an hour.
2. When a definition is redistributed (either manually by a user or automatically when a collection is changed)
3. When a previously unresponsive node responds to a set request for another definition. When the node responds, the Agent Policy Manager checks to see if there are any other outstanding distributions for that node and attempts to redistribute any it finds.

Viewing Currently Active Definitions

To view file monitoring or thresholding conditions that are currently active, select **View Current** from the APM main dialog. The window that is displayed shows a list of active definitions.

To view definition status for a particular node, select **Node Status** from the APM main dialog.

Viewing Pending Definitions

If you have distributed or partially distributed definitions, then modify the definition, you can view the modified distribution before distributing it. The modified definition has a status of PendingModifyDistribute. Select the **ViewPending** button to see the modified definition.

Deleting an Existing Threshold or File Monitor Definition

You can delete a threshold or file monitor definition through the APM dialog. On the APM main dialog, select the definition you want to remove, and click on **Delete**. A message is displayed in the APM EUI area to indicate that a delete operation is pending. Then select **Distribute** to distribute the delete operation to the nodes in the collection. This action removes the definition from the nodes. On the Distribute dialog that is displayed, select **Start** to delete the definition on all nodes. Messages appear to indicate the success or failure of the operation.

If any of the nodes cannot be reached, the delete operation has a status of PartiallyDeleted, even if none of the definitions could be deleted. If the problem was a timeout, such as if an agent was not available, you can redistribute manually by selecting **Distribute** again until all nodes are deleted. APM will continue to periodically redistribute the delete operation as well. The definition will remain in the database until APM is able to delete the definition from all nodes in the collection.

If a definition is in the PartiallyDeleted state, you can force deletion of the definition by selecting it and clicking on **Delete** again; however, this action may leave the definition still running in some of the nodes to which it has been distributed successfully. It may then be necessary to manually delete the definition from the hosts affected.

Forcing a deletion should only be done in cases where the hosts that have failed deletion requests are removed from the network and will not return to receive another deletion request.

To keep a definition, but remove the definition from all nodes in the network, do the following:

1. Select the definition on the APM main dialog.
2. Select **Modify**. The Modify Definition dialog is displayed.
3. Select all the collection names under Collection Assignments.
4. Select **Unassign** and click on **Apply**.
5. On the APM main dialog, the definition now has a status of PendingModifyDistribute. Select the definition and click on **Distribute**.

Defining Thresholds and File Monitor Conditions

This section explains how to fill in the fields on the Threshold and File Monitor dialogs in the APM EUI. If you are familiar with Systems Monitor, the dialogs will be familiar to you. They are very similar to the Threshold Table and File Monitor Table dialogs in the

Systems Monitor Configuration Application, with the added ability of assigning collections of objects to the definitions.

Defining Thresholds

Through the APM, you can collect important MIB data and set thresholds to send a trap or run a command when the threshold is tripped. Thresholds can be set on many types of MIB objects:

- MIB objects in the Systems Monitor SIA MIB
- MIB-I or MIB-II objects
- MIB objects in standard or private MIB extensions that can be retrieved using SNMP GETs (such as the MIB shipped with a router or other device)

With APM, you are setting a threshold against a collection of objects. The APM finds each of the objects in the collection and then determines which MLM in the network has monitoring responsibility for each of the objects. It will be an MLM in the same domain as the object. The APM then does an SNMP SET against the Threshold Table for each MLM to define the threshold. For example:

You define a collection `MYCOLL` that includes nodes A, B, C, and D.

Nodes A and B are in the same domain as Mid-Level Manager 1.
Nodes C and D are in the same domain as Mid-Level Manager 2.

Through APM, you define a threshold to be distributed to collection `MYCOLL`. When you select **Distribute**, an SNMP SET command is done against the Threshold Tables on Mid-Level Manager 1 and Mid-Level Manager 2. Each of these MLMs will begin polling their respective managed nodes for the MIB data specified in the threshold.

Note: Agent Policy Manager differs from the MLM in that you cannot prepend an alias or nodename to the beginning of the MIB object ID on which you are setting a threshold. Since you are setting thresholds against objects in a collection, there is no need for you to prepend a target to the front of the threshold variable.

To define a threshold through the APM interface, select **Threshold/Data Collection** using the Template button at the top of the APM main dialog.

The following dialog box is displayed:

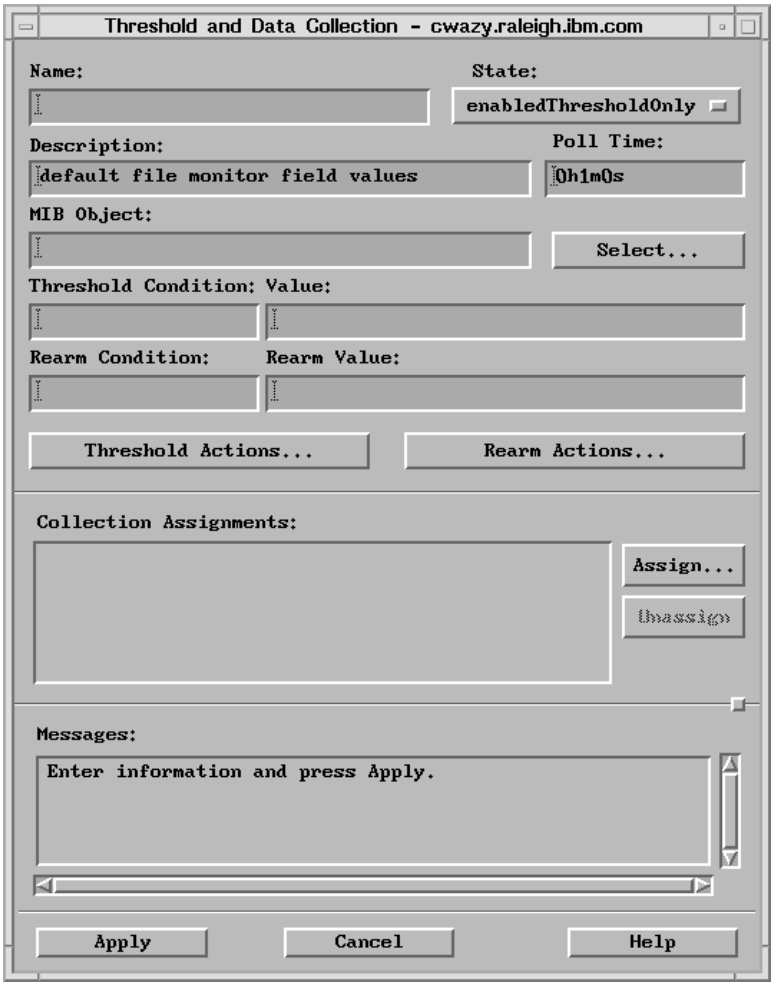


Figure 41. The Agent Policy Manager Threshold Definition Dialog Box

The fields in the table have the following meanings:

Table 16 (Page 1 of 4). Threshold Definition Dialog Fields

Field Name	Values	Purpose and Description
Name	Any unique character string.	**Purpose:** Specifies the name of the particular threshold entry. **Description:** The value in the Name field is used as a label and an instance ID, identifying each field for this threshold, and is appended to the corresponding object ID. The instance ID for each field in the same entry is the ASCII values for each letter of the name. For example, the value `APP_TIME` causes the instance ID for this entry to be `65.80.80.95.84.73.77.69`, where each letter is an ASCII integer representation of `APP_TIME`.

Table 16 (Page 2 of 4). Threshold Definition Dialog Fields

Field Name	Values	Purpose and Description
State	Valid values are: • disabled • enabledThresholdOnly • enabledStoreOnly • enabledThresholdStore	**Purpose:** Determines whether the threshold entry is active and what thresholding actions are being performed for this entry. **Description:** Their values and their meanings are described in the following list: • If the value is **disabled** the entry is valid, but the MLM will not check the value of the specified MIB variable against this threshold. • If the value is **enabledThresholdOnly**, the MLM is checking the retrieved MIB variable values against this threshold, then discarding the data. • If the value is **enabledStoreOnly**, the MLM is storing the retrieved MIB variable values in a file during the polling process; the values are not checked against the threshold. • If the value is **enabledThresholdStore**, the MLM is checking the retrieved MIB variable values against this threshold and storing the values in the collection file during the polling process.
Description	Any valid character string	**Purpose:** The general purpose of the threshold being performed. **Description:** Completing this field is not mandatory, but it is recommended. Describe the purpose and function of the threshold.
MIB Object	Any MIB variable in numeric, dot-notation format, followed by an instance ID in numeric or nonnumeric format. For multiple instances, a wildcard is valid.	**Purpose:** Indicates the MIB variables on which thresholding is to be performed. Do not prepend an alias or node address to the beginning of a MIB object. Since you are assigning the threshold to a collection, it is not necessary to specify a remote node. **Description:** You can specify the MIB variable in the following ways: • Specify one unique instance to be checked • Specify all instances of a variable For example, to perform threshold checking for the specific instance `interface 1` of the following MIB variable, `mib-2.interfaces.ifTable.ifEntry.ifInErrors` the value would be `.1.3.6.1.2.1.2.2.1.14.1`. To perform threshold checking for all instances of the previous MIB variable, the value is `.1.3.6.1.2.1.2.2.1.14.*`. You can use the alphanumeric string to indicate the instance to be checked. For example, to perform threshold checking for the MIB variable `sm6kSystemFileSystemPercentUsed` for the `/usr` directory instance, the value is `.1.3.6.1.4.1.2.6.12.2.5.2.1.4./usr.0`.

Table 16 (Page 3 of 4). Threshold Definition Dialog Fields

Field Name	Values	Purpose and Description
Poll Time	Valid values include an integer followed by one of these letters: • d = days • h = hours • m = minutes • s = seconds	**Purpose:** Indicates the time period that should elapse before the next threshold polling operation is invoked. If the letter is omitted, the default is minutes. For example, the value `1h10m` means to start this threshold polling operation `1 hour` and `10 minutes` following the last polling operation.
Threshold Condition and ReArm Condition	Valid condition values are: • = • < • <= • > • >= • & • | • changes • doesNotChange • exists • doesNotExist • !<*condition*>	**Purpose:** Indicates the condition used when checking retrieved MIB values against the threshold or rearm value. **Description:** • =, <, <=, >, and >= mean the retrieved value being checked against the threshold or rearm value must be either equal to, less than (or equal to), greater than (or equal to) the value, respectively. • Prepend ! to a condition to indicate what the value must **not** be. • **&** and **|** cause the retrieved values and threshold or rearm value to be combined with a logical AND or OR statement, respectively. A non-zero (0) result meets the condition; the trap or command operation is performed. • **changes** and **doesNotChange** cause watchdog operations to be performed. If a retrieved value changes or does not change, respectively, between consecutive polls, the condition is met. Threshold or rearm values are not used. • **exists** and **doesNotExist** check if the MIB variable exists. To check for an instance of the NetView for AIX subagent (trapgend daemon) within the MLM Process table, use `exists` and `.1.2.6.1.4.1.2.6.12.2.7.2.1.2.trapgend.*` as the variable. By default, thresholds for Counter type variables are computed by calculating the change per second in the sampled values and checking these delta values against the threshold or rearm value. For Gauge and Integer type variables, the actual variable value is checked against the threshold or rearm value. To override this default behavior, add the keyword **delta** or **value** in front of the condition, separating the keyword and condition by a space.
Value	Depending on the MIB variable, the value can a number, a floating point number, or a character string.	**Purpose:** Specifies the threshold or rearm value for a MIB variable against which retrieved values are checked. **Description:** This value must match the type of MIB variable that is being checked. If the MIB variable is numeric, use a number and a numeric check is done. If the MIB variable is a display string, and both the retrieved variable and this value can be converted to a floating point number, a floating point check is done. Otherwise, a string check is done. For example, if the MIB variable that is checked is a Counter type, the value should be an unsigned integer.

Table 16 (Page 4 of 4). Threshold Definition Dialog Fields

Field Name	Values	Purpose and Description
Specific or ReArm Specific	Any valid SNMP enterprise-specific trap number	**Purpose:** This field is on the Threshold Actions and Rearm Actions dialog boxes. It specifies the SNMP enterprise-specific trap which is to be sent when the threshold or rearm condition is met. If this value is 0 (zero), no trap is sent.
Enterprise or ReArm Enterprise	The valid SNMP enterprise ID, in dot notation, for the specified trap	**Purpose:** This field is on the Threshold Actions and Rearm Actions dialog boxes. It specifies the SNMP enterprise ID for the enterprise-specific trap which is to be sent when the threshold or rearm condition is met. If this field is blank, the Threshold Table enterprise ID is used.
Trap Description or ReArm Trap Description	Any valid character string. The description can use a set of defined environment variables.	**Purpose:** This field is on the Threshold Actions and Rearm Actions dialog boxes. It indicates the general reason for sending the trap. It is the first variable in the trap that is sent to the operator.
Command To Execute or ReArm Command To Execute	Any valid command can be specified. The command can use a set of defined environment variables.	**Purpose:** This field is on the Threshold Actions and Rearm Actions dialog boxes. It specifies a command to be executed by the MLM when the threshold or rearm condition is met. **Description:** The command is executed using the environment in which the MLM is running. If this field is blank, no command will be executed.

If you see the message `requested object name did not match the names available in the relevant MIB view`, it is possible that one of the target MLMs does not have the MLM MIB loaded, or the MLM agent (midmand) is not running.

Trap Destinations

You do not need to specify a destination for threshold traps, as you would need to for Systems Monitor. The trap destination is predefined to be all NetView for AIX managers that discover MLMs that are doing the thresholding and forwarding traps.

Monitoring Logs and Other Files

Through the APM, you can set up file and log monitoring on any node that has the Systems Monitor System Information Agent (SIA) installed. When you define a file monitor condition through the APM EUI and distribute it, SNMP SET commands are done against the actual SIAs running on the nodes defined in the collection for which you are defining a file monitor condition. Thus, the community names on the management node on which you are running the APM EUI on the remote SIA nodes must be defined so that the management station can do SNMP SETs on the SIA nodes.

Files of any type can be monitored, and you can monitor for many different conditions or combinations of conditions. Each condition returns a different trap. Conditions you can monitor for and the traps returned are as follows:

Condition Being Checked For	Trap Returned
Existence of a text string	Specific trap 21 - String found Specific trap 22 - File data modified
Changes to characteristics of the file, such as owner, group, or permissions	Specific trap 23 - File status changed
Existence of a file	Specific trap 24 - File does not exist Specific trap 25 - File exists

NetView for AIX has a predefined filter that will display only Agent Policy Manager file monitor traps in the Control Desk. You create a dynamic workspace and select the predefined C5filters filter to only display the traps. Step 9 on page 217 in “An Example of Defining and Distributing A Definition” on page 215 shows the steps for opening this dynamic workspace.

To define a file monitor condition through the APM interface, select **File Monitor** using the Template button at the top of the APM main dialog. The following dialog is displayed:

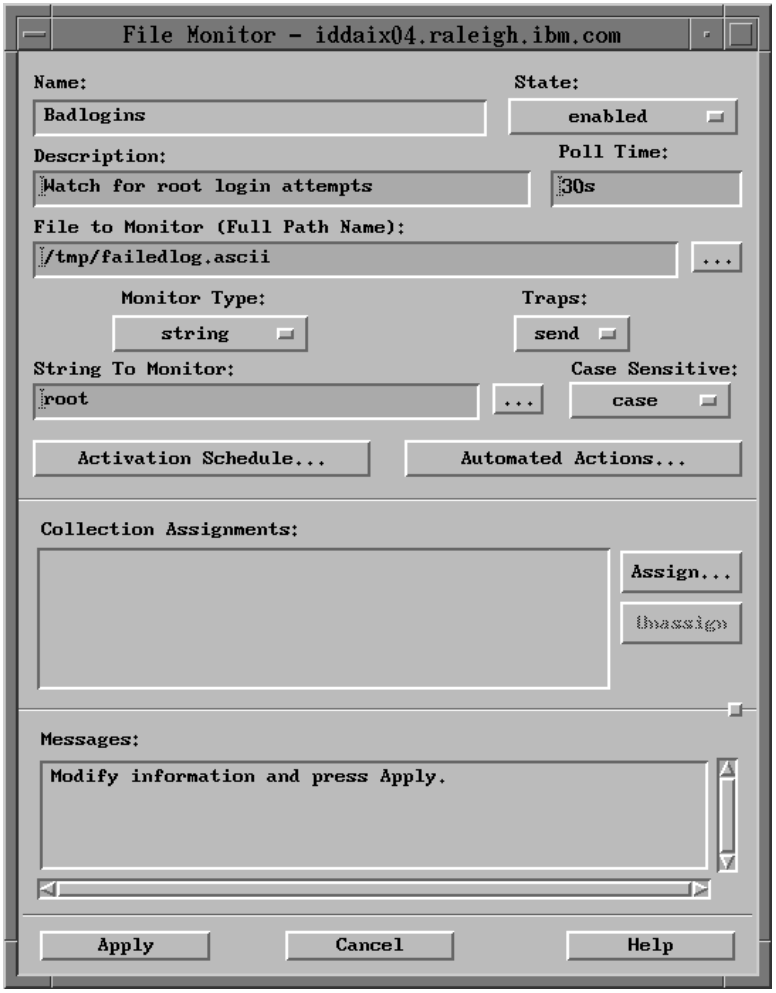


Figure 42. The Agent Policy Manager File Monitor Definition Dialog Box

The fields in the table have the following meanings:

Table 17 (Page 1 of 5). File Monitor Field Descriptions

Field Name	Values	Purpose and Description
Name	Any unique character string. No spaces are permitted in the string.	**Purpose:** Specifies the name to be used as a label and instance ID identifying each field for this command entry. The name is appended to the corresponding object ID. The instance ID for each field in the same entry is the ASCII values for each letter of the name. For example, a name of `RCMON` causes the instance ID for this entry to be `82.67.77.79.78`, which is an ASCII representation of the name `RCMON`.

Table 17 (Page 2 of 5). File Monitor Field Descriptions

Field Name	Values	Purpose and Description
State	Valid values are: • disabled • enabled • enabledFromBegin	**Purpose:** Specifies the current state of the file monitor table. **Description:** The values mean: • If the value is **disabled**, file monitoring is turned off. • If the value is **enabled**, file monitoring is turned on. Actual monitoring starts when the activation time you specified is reached. Systems Monitor begins searching at the end of the file, unless it determines that the file has wrapped or has decreased in size; if so, Systems Monitor starts searching at the top of the file. In this way, Systems Monitor uses only the newest file information. When the string you specified is found in the file, a trap is issued, and any action you specified in the Command to Execute field is run. • If the value is **enabledFromBegin**, file monitoring is active. Monitoring starts at the top of the file. You might specify this value if the file to be monitored is actually being generated each polling interval by a command that you specified to be run before each search, or if you always want the file to be searched from top to bottom each polling interval. Note that this state requires more CPU processing than the **enabled** state.
Description	Any valid character string	**Purpose:** States the general purpose of the file being monitored. **Description:** Describe the purpose of the monitored file and explain what action is taken when a search string or some other change to the file is detected.
Poll Time	Any valid character string	**Purpose:** Specifies the poll interval. **Description:** This variable specifies how often SIA should check the monitored file for the specified string. Specify the polling time as a decimal value and the unit of time. Units are: d=day, h=hour, m=minute, and s-second. If you do not specify a unit of time, the default is m=minute. Multiple units can be specified in this field. For example, you could specify a polling interval of 1h45m (1 hour and 45 minutes). If you do not specify a polling interval, the default is 10 seconds.

Table 17 (Page 3 of 5). File Monitor Field Descriptions

Field Name	Values	Purpose and Description
File to Monitor (Full Path Name)	Any valid character string	**Purpose:** Specifies the path name where the monitored file resides. **Description:** Specify the name of the file to be monitored, including the fully qualified path where the file resides. The file must be stored on the same node as the SIA MIB, and a **root** user must have read permission for the file. Monitoring of the file differs, depending on whether or not the file exists. • If the file already exists, monitoring begins with the *last* byte position in the file (the most recent records). • If the file does not exist, when you enable file monitoring, Systems Monitor waits until the file is created and begins searching for the string at the beginning of the file.
Monitor Type	Values are: • string • dataChange • statusChange • strDataStatus • notExist • exist • all	**Purpose:** Specify the type of monitoring to be performed. **Description:** This field specifies the type of monitoring to be performed on the specified string or condition. • If the value is **string**, the SIA watches the specified file for the string specified in the String to Monitor field. When the string appears, the specified actions are done. • If the value is **dataChange**, the SIA watches the file for any change to the contents of the file, such as added or deleted characters. If any data change occurs, the specified actions are done. • If the value is **statusChange**, the SIA watches the file for any change to the status of the file, including the file owner, file group, and file permissions. The file mode, user ID, and group ID that will be used for comparison are shown in the File Mode, User ID, and Group ID fields on the File Monitor window. The actual file mode, user ID, and group ID can be changed by modifying these fields in the File Monitor window. If the file mode, file owner, or file group changes, the specified actions are done. • If the value is **strDataStatus**, the SIA watches for any of the previous three types of changes (data changes, the appearance of a string, or file status changes). If any of these changes occur, the specified actions are done. • If the value is **notExist**, the SIA does the specified actions if the monitored file disappears (such as if it is erased). • If the value is **exist**, the SIA does the specified actions if the monitored file appears. • If the value is **all**, the SIA performs all types of monitoring (basically any change at all to the file).

Table 17 (Page 4 of 5). File Monitor Field Descriptions

Field Name	Values	Purpose and Description
Trap Selection	Valid values are: • send • noSend	**Purpose:** Specifies the types of traps that you want to have forwarded by the file monitoring function. **Description:** The values mean: • If the value is **send**, the file monitoring function sends all types of traps. • If the value is **noSend**, traps are not sent.
String to Monitor	Any valid character string	**Purpose:** Specifies the string to monitor for. **Description:** Specifies the file string that SIA should search for in the file. The string or pattern can be any limited regular expression (RE) in the style of the UNIX ed or egrep commands. Note that if you use anchor symbols (^ or $) in the expression, the string is only found if the line containing the anchor symbols is terminated with a new line (\n) character.
Case Sensitive	Valid values are: • case – Perform a case-sensitive search • ignoreCase – Ignore case on search	**Purpose:** Specifies whether the search is case sensitive.
Activation Schedule	Any valid character string for activation time	**Purpose:** specifies the time and days of the week when the file monitor condition is to be activated and deactivated. Activation – Use this field to delay the onset of file monitoring. Specify time in the format `HH:MM`, where `HH` is an integer between 0 and 23 and `MM` is an integer between 0 and 59. If this field is not set, the time is set to 00:00. Select the days of the week when you want file monitoring to be active. Deactivation Time – Use this field to set the time when file monitoring will be deactivated. Specify time in the format `HH:MM`, where `HH` is an integer between 0 and 23 and `MM` is an integer between 0 and 59. If this field is not set, the time is set to 00:00. Select the days of the week when you want file monitoring to be deactivated.

Table 17 (Page 5 of 5). File Monitor Field Descriptions

Field Name	Values	Purpose and Description
Automated Actions	Any valid character string	**Purpose:** The command to be run before monitoring begins, and the command to run if the string is found **Description:** Click on this button to set up commands to be run before file monitoring and during monitoring when a condition is found. Command to Execute Before Monitor – Indicates the command to be run before searches for the specified string or other change to the file are performed. This MIB variable can be used to generate, modify, or translate the file prior to the search. Command to Execute if Monitor Type Condition is Met – Use this field to specify a command to be run automatically when a certain string of text or some other kind of change occurs in the monitored file. For example, you could run a command to expand the size of a file system if you are monitoring for a message that the file system is full. Indicates the command to be run before searches for the specified string or other change to the file are performed.
Collection Assignments		Use this field to assign collections to this definition. Click on the **Assign** button to open a dialog box for selecting collections.

Diagnosing Problems Using the Problem Determination Assistance Facility

Agent Policy Manager provides a tool that can aid you in determining what the conditions were that triggered a threshold or file monitor trap. To see the PDA menu, do the following:

1. Double-click with the left mouse button on the Agent Policy Manager Monitors icon. A map with defined thresholds and file monitor conditions is displayed.
2. Double-click on a threshold or file monitor icon to see the objects in the collection.
3. Double-click down to the node-level view.
4. If a threshold or file monitor condition was met, one of the icons is red. If the session between the node and its managing MLM went down, the SM Map icon is red. Double-click on the icon with the left mouse button. The PDA menu is displayed. (Note that you can start the PDA menu even if all the icons are green.)

From this menu, you can see a summary of the file monitor or threshold condition settings, start some applications to help you diagnose the problem that triggered the threshold or file monitor condition, and reset the color of file monitor icons.

Resetting the Color of a File Monitor Icon

You can reset the color of a file monitor icon from the PDA menu by clicking on the **Reset** button in the middle of the dialog. The color of the icon from which you started the PDA menu is changed to green, and the change in status is propagated up to the

root map. If there are other icons you want to reset, you must start the PDA menu from each icon to reset the color.

Applications Available from the PDA Screen

You can select the following diagnosis applications from the PDA Diagnosis and Corrections menu:

- **Display submap** starts a submap showing the node-level view for the affected node. This view has several icons.
  - SM Map appears on node level views of MLM collection nodes and nodes that have thresholds set against them. Its color indicates the status of the connection between the node and the MLM that is managing it.
  - tr0 is the node's interface card, as on other NetView for AIX maps.
  - One or more executable icons indicate thresholds and file monitor conditions that have been created.
- **Dynamic events for this node** opens a dynamic workspace showing all events for the node.
- **MIB Browser** starts the xnmbrowser application for this node.
- **Historic Graph Data** starts the xnmgrapher application and graphs the data that was stored for thresholds against the node.

By clicking on the **Add** button, you can add the following applications to the PDA menu:

- **Historic events for this node** opens a static workspace showing all events for the node.
- **Print** starts the NetView for AIX Print Tool application.
- **Mail** starts the mail application.

Agent Policy Manager Reference

The following sections have more detailed information about the Agent Policy Manager and how it interacts with Systems Monitor and the network.

Configuring Community Names

The APM application uses SNMP community names to control the access the NetView for AIX manager has to managed objects in the network. Community names are passed back and forth in SNMP GET and SET requests and in traps. The community name is similar to a password in that it determines whether an entity can gain access to information or perform an action.

On AIX network devices, SNMP community names are defined in a file called **snmpd.conf**. The AIX SNMP agent, snmpd, checks this file when it receives a request for information.

NetView for AIX and its associated applications, such as Systems Monitor, use another file called **ovsnmp.conf** to associate objects with community names. NetView for AIX and the Systems Monitor MLM check this file when the send requests for information.

In summary:

- When a node *sends* a request, it uses the **ovsnmp.conf** file to determine what community name should be sent in the request. The community name associated with the target node is included in the request.
- When a node *receives* a request, it uses the **snmpd.conf** file to validate the community name sent by the requester. The community name associated with the requesting node is compared with the name included in the request.

The community name sent by the requester node and the community name expected by the receiving node *must match* for the request to be executed as requested. If they do not match, the receiving node will use a community name of **public** by default.

Mismatched community names cause authentication errors. These errors will in turn prevent the distribution of an APM file monitor or threshold definition from being successful.

You should not edit the **ovsnmp.conf** file directly. Use **Options..SNMP Configuration** from the NetView for AIX menu bar to set the community name to be sent to the target node.

If you are defining a threshold condition in your network, your NetView for AIX manager station must have SNMP SET access to the nodes running MLMs. For this reason, it might be easier to use a single community name in your network. The NetView for AIX security feature can provide the application security that you require. See Chapter 2, “Defining and Managing a Security Policy” on page 19 for more information.

Community Name Examples

Here are two scenarios of how entries in the ovsnmp.conf and snmpd.conf files change, depending on which objects you are monitoring. For simplicity, these examples show a single machine running each agent. In a real working network, community names would be set up on a much wider basis, such as across a subnet or even across the whole network.

First Example

NetView for AIX, an MLM, and an SIA are on a machine called `nvmgr`. You want to set a thresholding condition against that MLM and monitor a log file using the SIA file monitoring function.

- The **ovsnmp.conf** file is edited using the SNMP Configuration dialog from the NetView for AIX Options menu:

```
# MLM entry (LOOPBACK)
127.0.0.1:system:*:::::system:
# SIAentry
nvmgr:siaset:*:::::siaset:
```

- The **snmpd.conf** file entries are entered as follows:

```
# MLM name
community       system  127.0.0.1 255.255.255.255 readWrite
# SIA name
community       siaset  127.0.0.1 255.255.255.255 readWrite
```

Second Example

An MLM and an SIA are on a machine called `workstation`. From the `nvmgr` node, you want to set a thresholding condition against that MLM and monitor a log file using the SIA file monitoring function.

- The **ovsnmp.conf** file is edited using the SNMP Configuration dialog from the NetView for AIX Options menu:

```
# MLM entry
workstation:mlmset:*:::::mlmset:
# SIAentry
workstation:siaset:*:::::siaset:
```

- The **snmpd.conf** file entries are entered as follows:

```
# MLM name
community       system  127.0.0.1 255.255.255.255 readWrite
# SIA name
community       siaset  127.0.0.1 255.255.255.255 readWrite
```

See the *Systems Monitor User's Guide* for extensive scenarios showing how you would configure community names depending on where you have the Systems Monitor features installed.

Agent Policy Manager Aliases

The Systems Monitor MLM MIB includes an Alias Table that is used for several of the MLM's functions. If the administrator configures the MLMs in the network to take over local discovery and status monitoring duties from the NetView for AIX manager, the MLM assigns aliases for groups of nodes to facilitate management of the nodes. The APM also use the Alias Table to keep track of groups of nodes. It sets aliases for collections that have thresholds set against them, and for the thresholds themselves.

After you define a threshold and distribute it to a collection, an alias is defined for this collection. Aliases for collections have names in the format `{NVip-address_collname}`, where `ip-address` is the IP address of the NetView for AIX manager, and `collname` is the name of the collection. For example, a collection called Fileservers assigned to the MLM by the NetView for AIX manager on address 9.67.102.16 would be assigned an alias of {NV9.67.102.16_Fileservers}.

Similarly, there is an alias for each threshold you define. Thresholds have aliases in the format `{NVip-address_thresholdname}`, where `ip-address` is the IP address of the NetView for AIX manager, and `thresholdname` is the name of the threshold.

Managing Aliases

The alias definitions are APM's mechanism for managing changes to collections and thresholds. Collections are dynamically updated to reflect changes in a network. When a collection is changed, the change is made to the alias definition as well. You do not have to redistribute a threshold definition if you use the collection editor to change the members of a collection; the APM automatically redistributes, based on the changes to the alias.

Agent Policy Manager MLM Domains

For thresholding, APM defines domains for the MLMs in the network. Basically, it determines where the MLMs are and divides the managed objects as equitably as possible.

Initially, MLM domains are defined based on subnet IDs. Nodes that are in the same subnetwork as an MLM are assigned to that MLM. Nodes that do not have an MLM in their subnet are assigned to a default domain. There is no overlap in these initial definitions; each node is managed by one MLM only.

You can see these domains from the NetView for AIX root map. The MLM domains are under the icon **MLM Managers**. Also, if you start the Collection Editor, you will see the MLM domains (as well as the mlmDomain_Default domain) listed as already existing collections.

APM chooses the MLM that will serve as the default domain depending on how your network is set up:

1. If there is an MLM on the NetView for AIX manager, it will be used.
2. If the MLM is not installed on the NetView for AIX manager, you must assign the default domain collection to one of the MLMs in your network using the Collection Editor.

If there is only one MLM in your network, that MLM is assigned all nodes, and a default domain might not be created.

Rearranging MLM Domains Manually

You might want to alter the distribution of nodes to MLMs to facilitate your management of the network. For example, you might want an MLM to manage all the routers in the network, or to manage all objects in a physical location. To do this, start the Collection Editor and select the domain you want to change, then edit the list of nodes. Since the MLM domain is just a collection, you can use the same rule logic that you use for defining collections to set thresholds against. You do not need to reassign any nodes that were in the old domain but are not in the new domain; Agent Policy Manager takes care of assigning new domain responsibilities for the nodes.

If you have MLMs that are offloading discovery and status monitoring from NetView for AIX and are very busy as a result, you might want to lighten their workload by moving nodes out of their APM domains.

If you add a new MLM to your network, Agent Policy Manager will recognize it as an MLM and assign a domain collection, but the collection will be empty. If you edit this collection and add nodes to it, the Agent Policy Manager will take care of moving node responsibilities accordingly.

Rearranging MLM Domains Automatically

If an MLM disappears from the network, the Agent Policy Manager will automatically redistribute that MLM's workload. A new collection is created called `delDomain_mlmname`, where `mlmname` is the name of the MLM. The nodes for which the MLM was responsible are redistributed to other MLMs in the network. They are assigned according to their subnets.

If the MLM later reappears in the network, the domain is recreated and the nodes are reassigned to the original MLM.

Components of the Agent Policy Manager

Think of APM as having two main pieces:

- The APM daemon
- The APM interface

The APM daemon

The APM daemon is the “glue” that binds together Systems Monitor, NetView for AIX, and the Collection Facility. This daemon performs several functions:

- It receives definitions from the APM Configuration Interface and makes appropriate changes to Systems Monitor configuration.
- It updates the APM interface to report changes in status.
- It handles incoming Systems Monitor threshold and file monitoring traps and updates NetView for AIX submaps and nvevents accordingly.
- It catches update information from the Collection Facility and makes changes to threshold and file monitor definitions on nodes that were added to or deleted from collections.
- It retries failed definitions, both when the daemon is started and periodically thereafter.

When the daemon is started, it searches the object database, looking for MLMs. It creates domains for all MLMs and assigns one to be the default domain. If this is not the first time the Agent Policy Manager has been started, it will look for new MLMs and create domains for them, and it will move domain responsibilities for any deleted MLMs to another MLM that is still active.

At startup, the daemon also retries any outstanding distributions, including any partially distributed or failed definitions and deletions, and any distributions or deletions that were in progress when the daemon stopped.

The APM Configuration Interface

The APM interface is your tool for defining and manipulating thresholding and file monitoring definitions. Much like the Systems Monitor Configuration Application, the APM configuration interface gives you fields for defining information about a threshold or file monitoring condition. Through the APM configuration interface you can get a list of conditions that have been defined, add or delete definitions, and distribute definitions. You specify collections to which a definition will be applied through the EUI.

Distribution Status Indicators

Table 18 (Page 1 of 2). Status Indicators for Distribution Agent Policy Manager Definitions

Status	Meaning
NeverDistributed	An attempt was never made to distribute this definition. Click on **Distribute** to distribute the collection.
Distributed	The definition was successfully distributed to all nodes in the collection.
PartiallyDistributed	An attempt was made to distribute the definition, but one or more nodes could not be modified. It could be that none of the nodes were reached. If the failure is due to a timeout, you might want to let Agent Policy Manager continue to redistribute on its own.
PartiallyDeleted	An attempt was made to delete the definition, but one or more nodes could not be modified. It could be that none of the nodes were reached. Click on **Delete** to retry the deletion. Click on **Undo** to discontinue the operation. You can also do nothing and allow Agent Policy Manager to continue to try to distribute the deletion.
PendingModifyDistribute	A modification was made to the definition, but it has not yet been distributed. Click on **Distribute** to distribute the definition.
PendingDeleteDistribute	The user selected a definition in the APM main dialog and clicked on **Delete**. Select **Distribute** to delete the definition on all nodes and delete the definition. Select **Undo** if you do not want to delete the definition.
DistributeInProgress	Agent Policy Manager was doing a distribution when the C5d daemon went down or a logic error occurred. If you see this error after C5d recycles, allow the C5d daemon to continue to distribute. If you are seeing several items with this status, recycle the daemon.
ModifyDistributeInProgress	Agent Policy Manager was distributing a modified definition when the C5d daemon went down or a logic error occurred. If you see this error after C5d recycles, allow the C5d daemon to continue to distribute. If you are seeing several items with this status, recycle the daemon.

Table 18 (Page 2 of 2). Status Indicators for Distribution Agent Policy Manager Definitions

Status	Meaning
DeleteDistributeInProgress	Agent Policy Manager was distributing a deletion when the C5d daemon went down or a logic error occurred. If you see this error after C5d recycles, allow the C5d daemon to continue to distribute. If you are seeing several items with this status, recycle the daemon.

Appendix. NetView for AIX Internal Traps

Use this reference to understand the traps that are generated by NetView for AIX.

Terms and Conventions

The following terms and conventions are used:

- Words in italics are explanatory variable names. They are replaced by the value of the variables at run time when the event occurs.
- The term event is used interchangeably with the term trap.
- The term event log generally implies either the trapd.log file or the event card application display.
- Time stamp values are displayed in epoch time. Epoch time is the number of seconds elapsed since an *epoch*, which in AIX is January 1, 1970.
- The term objid is an OVw database object ID.
- Link Level Address, (LLA) is the same as physical address which is the physical address of the interface cards. The LLA is typically displayed as a 6 byte, (or 12-digit hex) number.
- When a demand poll is used as a method of correction, it should be pointed out that a poll is, by default, performed once-a-day for each node. Performing a demand poll is a method of effecting immediate action.

  Note: A demand poll can be performed from a separate window by the command **nmdemandpoll** *nodename* where node name is the node name as it exists in the topology database (generally the fully qualified DNS name for the node).

Internal NetView for AIX Traps

The only trap that affects the operation of NetView for AIX is the SNMP_EV (58916871) trap, which can be generated to facilitate Configurable Status.

All NetView for AIX traps are generated with the enterprise ID `.1.3.6.1.4.1.2.6.3.1` and the generic number of 6, which means enterprise-specific. The specific number for each trap is listed in Table 19 on page 243. A NetView for AIX trap is generated with 5 variable bindings. The OID for each varbind is also listed, though it has relatively no importance in the generated trap.

Varbind 1 MIB OID: .1.3.6.1.4.1.2.6.3.1.1.2.0

This variable is the source ID. It is an integer value that corresponds to the internal component of the NetView for AIX that generated the event. Following are the separate Source ID values and their corresponding components. Each source ID value has a unique letter (identified in parenthesis) that is used for identification in the trapd.log file. For

instance, the Node Up event text will be preceded by an N, indicating that the source for the trap was the netmon application.

```
(A)gent                (L)oad MIB          (s)ppld
(a)pplication          (N)etmon            (T)rapd
(D)ata Collector       NonI(P) Topology    t(r)alertd
(d)emo/LoadHosts       (n)etmon related    (V)endor
(E)vent Application    (O)SI_SuperAgent    xnm(C)ollect
(I)PMAP_SuperAgent     (M)ap/ovtopmd       xnm(t)rap
(i)gnore               (S)ecurity Agent
```

Varbind 2 MIB OID: .1.3.6.1.4.1.2.6.3.1.1.3.0

This variable is a string value identifying the host name to which this trap applies. If there is no applicable node, <none> is displayed. As in our previous varbind example, if the Node Up event is generated, then varbind-2 will contain the host name of the node that was detected as operational.

Varbind 3 MIB OID: .1.3.6.1.4.1.2.6.3.1.1.4.0

This variable is a string value containing a description of the event that was generated. For the Node Up event, varbind-3 would contain the text `Node Up`. This varbind is most important because the 3rd varbind is the text that gets displayed in the event log. This is most evident by looking at the Event Configuration screen of the product. For each NetView for AIX, the corresponding event log information contains $3. This indicates that the 3rd varbind's contents are to be displayed in the event log. The exact text for each trap is documented in Table 19 on page 243.

Varbind 4 MIB OID: .1.3.6.1.4.1.2.6.3.1.1.5.0

This variable is a string value containing internal data that is particular to the trap. This field generally contains data such as Time stamp values, object ID values for node and interface objects, IP addresses, and so forth. This information generally has no meaning to the user. The exact value for each trap is documented in the tables.

Varbind 5 MIB OID: .1.3.6.1.4.1.2.6.3.1.1.6.0

This variable contains a string value containing the database name. This name must be `openview`.

Trap list

The traps are listed in numerical order. For each trap listed, the following information is provided:

- Specific trap number
- Trap name
- Trap description
- Description field (descr). This contains the contents of varbind-3.
- Data field. This contains the contents of varbind-4.
- Condition. This indicates the condition that causes the trap to be generated.

Traps generated by NetView for AIX are described as follows.

Table 19 (Page 1 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
50462720 WARN_EV Warnings	Descr field	It can be one of the following strings. The number here corresponds with the number under the Condition field. 1. WARNING Attempted addition of object with no object ID, *objectName* 2. WARNING Attempted addition of existing object *objectId*, returning error 3. WARNING could not allocate object ID for segment *segName* 4. WARNING Unset field values with object id *oid* failed: OVwError = *errorString*. 5. WARNING invalid SNMPTrap packet from agent *addr* source *sourceId* pid *pid*
	Data field	NULL
	Condition	Following are the conditions: 1. Attempt was made to add an object with null object ID to the topology database. 2. Attempt was made to add an already existing object to the topology database. 3. Creation of object ID for segment fails. 4. Unsetting of field values in the topology database fails during cleanup. 5. Error parsing an SNMP trap from an agent.
50790400 NM_EV Node Marginal	Descr field	Node Marginal
	Data field	Contains the following: • Time stamp • objid of the node object
	Condition	Node status change to a marginal state.
50790401 SN_EV Segment Normal	Descr field	Segment *segName* Up.
	Data field	Contains the following: • Time stamp • objid of the segment object
	Condition	Segment status changes to normal.
50790402 SM_EV Segment Marginal	Descr field	Segment *segName* Marginal
	Data field	Contains the following: • Time stamp • objid of the segment object
	Condition	Segment status changes to marginal.

Table 19 (Page 2 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
50790403 NETN_EV Network Normal	Descr field	Network *netName* Up
	Data field	Contains the following: • Time stamp • objid of the network object
	Condition	Network status changes to normal.
50790404 NETM_EV Network Marginal	Descr field	Network *netName* Marginal
	Data field	Contains the following: • Time stamp • objid of the network object
	Condition	Network status changes to marginal.
50790405 SA_EV Segment Added	Descr field	Segment *segName* Added
	Data field	Contains the following: • Time stamp • objid of the segment object
	Condition	A segment was added.
50790406 SD_EV Segment Deleted	Descr field	Segment *segName* Deleted
	Data Field	Contains the following: • Time stamp • objid of the segment object
	Condition	A segment was deleted.
50790407 NETA_EV Network Added	Descr field	Network *netName* Added.
	Data field	Contains the following: • Time stamp • objid of the network object
	Condition	A network was added.
50790408 NETD_EV Network Deleted	Descr field	Network *netName* Deleted.
	Data field	Contains the following: • Time stamp • objid of the network object
	Condition	A network was deleted.
50790411 CPP_EVChange Polling Period	Descr field	Polling Intervals changed
	Data field	NULL
	Condition	Polling intervals were changed via the Topology/Status Polling Intervals window. This trap informs netmon that it needs to re-read the polling intervals file.

Table 19 (Page 3 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
50790412 FP_EV Forced Poll Event	Descr field	Demand polling on node *nodename*
	Data field	Contains the following: *pipename* The pipe opened for communication with netmon *debugflag* Reserved for future use
	Condition	When demand-poll is initiated by nmdemandpoll. The application communicates to netmon by sending this event.
50790416 MNET_EV Manage Network	Descr field	Network *netname* Managed
	Data field	Contains the following: • Time stamp • objid of the network object
	Condition	Event generated upon managing a network.
50790417 UNET_EV Unmanage Network	Descr field	Network *netname* Unmanaged
	Data field	Contains the following: • Time stamp • objid of the network
	Condition	Event generated upon unmanaging a network.
50790418 MN_EV Manage Node	Descr field	Node Managed
	Data field	Contains the following: • Time stamp • objid of the node
	Condition	Event generated upon managing a node.
50790419 UN_EV Unmanage Node	Descr field	Node Unmanaged
	Data field	Contains the following: • Time stamp • objid of the node
	Condition	Event generated upon unmanaging a node.
50790420 MSEG_EV Manage Segment	Descr field	Segment *segname* Managed
	Data field	Contains the following: • Time stamp • objid of the segment
	Condition	Event generated upon managing a segment.
50790421 USEG_EV Unmanage Segment	Descr field	Segment *segname* Unmanaged
	Data field	Contains the following: • Time stamp • objid of the segment
	Condition	Event generated upon unmanaging a segment.

Table 19 (Page 4 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
50790423 NMTM_EV Netmon Change Trace Mask	Descr field	Netmon Trace mask changed.
	Data field	*tracemask*
	Condition	If the **-M** option is used to change netmon's tracemask, then this trap will be generated to inform the netmon daemon of the new tracemask value.
50790427 CIS_EVChange Interface Segment	Descr field	One of the following fields: • Interface *iflabel* no longer connected to segment. • Interface *iflabel* transferred to segment *segName* • Interface *iflabel* Transferred to segment *segId*
	Data field	Contains the following: • IP address • Time stamp • Node objid • Interface objid • Segment objid
	Condition	A change of interface event.
50790438 FMTCHG trapd.conf file format change	Descr field	*progname* changed format file /usr/OV/conf/C/trapd.conf.
	Data field	NULL
	Condition	Generated when contents of the trapd.conf are changed either by the graphical interface application, xnmtrap, or the command line interface, addtrap. This informs any interested applications that they need to re-read the trapd.conf file.
50790439 MIBCHGASN.1 mib definition file format changed	Descr field	New MIB library file.
	Data field	NULL
	Condition	Generated when a MIB is loaded with the MIB Compiler/Loader. This informs any interested applications that a new MIB binary file is available.
50790440 COLCHGSNMP data collector file format changed	Descr field	SNMP data collector started.
	Data field	xnmcollect changed format file /usr/OV/conf/snmpCol.conf
	Condition	Generated when the SNMP Data collector configuration file is updated. This informs the snmpCollect daemon that it needs to re-read its configuration file.
50790441 MI_EV Manage Interface	Descr field	Interface *iflabel* managed.
	Data field	Contains the following: • IP address • Time stamp • Node and interface objids
	Condition	Generated upon managing an interface.

Table 19 (Page 5 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
50790442 UI_EV Unmanage Interface	Descr field	Interface *iflabel* unmanaged.
	Data field	Contains the following: • IP address • Time stamp • Node and interface objids
	Condition	Generated upon unmanaging an interface.
50790443 NETFC_EV Network Flags changed	Descr field	Network *netName* has new flags: *netFlags*.
	Data field	Contains the following: • Time stamp • Network object ID • New network flags • Old network flags
	Condition	Event noting change of internal flags for network. Flags are: `REMOVED USER_ADDED LAYOUT_OFF USE_XY SERIAL_NETWORK`
50790444 SEGFC_EV Segment Flags changed	Descr field	Flags for Segment *segName* changed to *segNewFlags*
	Data field	Contains the following: • Time stamp • Segment objid • New segment flags • Old segment flags

	Condition	Event noting change of internal flags for segment. Flags are:
		`REMOVED USER_ADDED LAYOUT_OFF USE_XY BUS_SEG`
		`STAR_SEG TOKEN_RING FDDI_RING SERIAL_SEG`

50790445 NFC_EV Node Flags changed	Descr field	Node Flags changed to *nodeFlags*
	Data field	Contains the following: • Time stamp • Node object ID • NewNodeFlags • OldNodeFlags

	Condition	Event noting change of internal flags for node. Flags are:
		`REMOVED USER_ADDED LAYOUT_OFF USE_XY CONNECTOR`
		`GATEWAY STAR_HUB IS_SMART_CONN IS_BRIDGE`

Table 19 (Page 6 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
50790446 IFC_EV Interface Flags changed	Descr field	Flags for Interface *ifLabel* changed to *newIfFlags*
	Data field	Contains the following: • IP address • Time stamp • Node objid • Interface objid • NewIfFlags • OldIfFlags

	Condition	Event noting change of internal flags for interface. Flags are:
		`REMOVED  USER_ADDED`
		`LAYOUT_OFF  USE_XY`
		`CONNECTED_TO_NET`
		`CONNECTED_TO_SEG`
		`SEGMENT_HUB`
		`NOT_CONNECTED`

58720256 CPUL_EV CPU Load	Descr field	CPU Load
	Data field	Value for the CPU usage index.
	Condition	If the CPU Load usage index exceeds a specified threshold, then this event gets generated. This is obsolete functionality. The SNMP Data Collector now performs all thresholding.
58720257 DSPU_EV Disk Space Percentage Used	Descr field	Disk space percentage used *string*
	Data field	Disk usage
	Condition	If the disk space usage exceeds a specified threshold, then this event gets generated. This is obsolete functionality. The SNMP Data Collector now performs all thresholding.
58720258 IPD_EV Interface Percent Deferred	Descr field	Interface percent Deferred *interfaceAddr*
	Data field	New value.
	Condition	If the percent packets deferred exceeds a specified threshold, then this event gets generated. This is obsolete functionality. The SNMP Data Collector now performs all thresholding.
58720259 IPC_EV Interface Percent Collisions	Descr field	Interface Percent Collisions *interfaceInetAddr*
	Data field	New value.
	Condition	If the percent collision packets exceeds a specified threshold, then this event gets generated. This is obsolete functionality. The SNMP Data Collector now performs all thresholding.
58720260 ICE_EV Interface CRC Errors	Descr field	Interface CRC Error interfaceInetAddr
	Data field	New value.
	Condition	If the number of CRC errors exceeds a specified threshold, then this event gets generated. This is obsolete functionality. The SNMP Data Collector now performs all thresholding.

Table 19 (Page 7 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
58720261 IPIE_EV Interface Percent Input Errors	Descr field	Interface Percent Input Error *interfaceInetAddr*
	Data field	New Value
	Condition	If the percent input errors exceeds a specified threshold, then this event gets generated. This is obsolete functionality. The SNMP Data Collector now performs all thresholding.
58720262 IPOE_EV Interface Percent Output Errors	Descr field:	Interface Percent Output Errors *interfaceInetAddr*
	Data field	New value
	Condition	If the percent output errors exceeds a specified threshold, then this event gets generated. This is obsolete functionality. The SNMP Data Collector now performs all thresholding.
58720263 DCOL_EV Data Collector Detected Threshold	Descr field	*mibAlias instance* threshold exceeded *(>thresholdValue): snmpValue*
	Data field	mibName
	Condition	The SNMP Data Collector will generate this event when it detects that a threshold MIB value has been exceeded.
58720264 DCRA_EV Data Collector re-arm event	Descr field	*mibAlias instance* threshold rearmed *(<=resetValue): snmpValue*. Sampled high of *highValue* at *highTime*
	Data field	*mibName*
	Condition	The SNMP Data Collector will generate this event when it detects that a thresholded MIB variable has descended below the rearm value. The MIB variable is then rearmed for the threshold event, DCOL_EV.
58785792 IADD_EV Interface Added	Descr field	It can be one of the following strings: • Interface *ifLabel* Added • Connection Added to *segmentName* • Connection Added
	Data field	Contains the following: • IP address • Time stamp • Node and interface objids
	Condition	netmon has detected a new interface -- from a successful ping.

Table 19 (Page 8 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
58785793 IDEL_EV Interface Deleted	Descr field	It can be one of the following strings: • Interface *ifLabel* Deleted • Connection Deleted to *segmentName* • Connection Deleted
	Data field	Contains the following: • IP address • Time stamp • Node and interface objids
	Condition	The interface is to be deleted, either from user interaction, or from netmon.
58785794 NADD_EV Node Added	Descr field	Node Added
	Data field	Time stamp and node objid
	Condition	The node is added to the topology database.
58785795 NDEL_EV Node Deleted	Descr field	Node Deleted
	Data field	Contains the following: • Time stamp • Node objid
	Condition	The node is removed from the topology database.

Table 19 (Page 9 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition

58851329
ERR_EV
Non-Fatal errors

Most of these are internal debugging messages. The “Agent in distress” messages have been seen with misbehaving agents.

Descr field — It can be one of the following strings. The number here corresponds with the number under the Condition field.

1. ERROR Internal Error - object of bad type *objectType* detected in addObj
2. ERROR Internal Error - object of bad type *objectType* detected in remove
3. ERROR Internal Error - object of bad type *objectType* detected in delete
4. ERROR Could not create directory *directory: errorNumber*
5. ERROR Could not create main database directory: *errorNumber*
6. ERROR Could not allocate object ID for interface *ifName*
7. ERROR Could not allocate object ID for net *netName*
8. ERROR Could not allocate Topo Object ID for Node *nodeName*
9. ERROR Unable to format time in formatDBtime, return value = *retVal*
10. ERROR Could not look up field id for field *ovwNselectionName*
11. ERROR Could not get unique object name for *baseName*, ovw_error = *ErrorString*
12. ERROR Could not allocate nor lookup object ID for *objectName*
13. ERROR Internal Error: too many fields for fieldBindList!
14. ERROR SetFieldValues returned *returnedValue* for obj *objectId*, OVwError = *errorNumber: errorString*.
15. ERROR Too many fields to unset, skipping extra
16. ERROR Could not look up field id for name *ovwFIPAddress*
17. ERROR Too many values for ListFieldValue
18. ERROR Could not look up field id for name field *ovwFIPNetworkName*
19. ERROR Could not look up field id for name field *ovwFIPHostName*
20. ERROR delNetworkCmd: Error removing net *netName* from topology: *topoErrString*
21. ERROR delNetworkCmd: Error removing seg *segName* from topology: *topoErrString*
22. ERROR Demand poll: cannot stat pipe *pipename*
23. ERROR Demand poll: *pipename* not a pipe
24. ERROR Pipe open failure *pipename* (errno = *errorNumber*)
25. ERROR Pipe fdopen failure *pipename* (errno = *errorNumber*)
26. ERROR netmon stopped -- dumping list
27. ERROR Fatal error in *placeOfError* logged in trace file - exiting
28. ERROR netmon: stopping in timeout_snmps
29. ERROR Agent in distress: spinning in ipRouteTable on *nodeName*: Aborting request!
30. ERROR Agent in distress: spinning in ifTable on *nodeName*: Aborting request!
31. ERROR Agent in distress: spinning in ipAddrTable on *nodeName*: Aborting request!

Table 19 (Page 10 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition
58851329	
Data field	NULL
Condition	Following are the conditions: 1. The program enters in default case in the function addObjToTopo(). 2. The program enters in default case in the function removeObjFromTopo(). 3. The program enters in default case in the function deleteObjFromTopo(). 4. Creation of directory fails. 5. Creation of /usr/OV/databases/<dbname> fails. 6. Creation of object ID for an interface fails. 7. Creation of object ID for a network fails. 8. Creation of object ID for a node fails. 9. Conversion of time format between system and INGRES/SQL formats failed. 10. Failed to get the field id from selection name. 11. Failed to get unique object name from selection name ID and base name. 12. Either failed to create object id or failed to get an object ID for an already existing object. 13. An attempt is made to increase the number of fields in the bind list when it is already full. 14. Unable to set the value of the fields in the bind list. 15. The number of maximum topology specific fields which an object can have is MAX_UNSET_FIDS. During cleanup if more than 50 fields are found this event is sent. 16. Failed to convert a field name to field ID. 17. When an attempt is made to add another value to a field which is of list type and already the field has maximum number of permissible values. 18. Failed to get field ID from field name. 19. Failed to get field ID from field name. 20. Failed to remove network from topology. 21. Failed to remove segment from topology. 22. 'Stat' call fails on pipe. 23. A pipe descriptor was expected but it is not a pipe. 24. Failed to create a pipe. 25. Failed to open a pipe (fdopen call failed). 26. Core dump has occurred while running netmon. 27. Fatal error has occurred. 28. The SNMP wait queue has nodes to process, yet the node contains no state information. 29. The specified node is spinning in its ipRouteTable. Aborting requests. 30. The specified node is spinning in its ifTable. Aborting requests to the node. 31. The specified node is spinning in its ipAddrTable. Aborting requests.

Table 19 (Page 11 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
58851330 FERR_EV Fatal Errors	Descr field	It can be one of the following strings. The number here corresponds with the number under the Condition field. 1. FATAL ERROR Out of memory -- exiting. 2. FATAL ERROR Could not map field *fieldName* into OVwFieldId 3. FATAL ERROR><Node/Iface or memory allocation error in *where* - exiting 4. Netmon probably died: ungracefully disconnected from trapd 5. snmpCollect probably died: ungracefully disconnected from trapd 6. topmd probably died: ungracefully disconnected from trapd 7. *applicationName* reached maximum number of outstanding events, disconnecting from trapd. 8. Application reached maximum number of outstanding events, disconnecting from trapd.
	Data field	NULL
	Condition	Following are the conditions: 1. When allocation of dynamic memory fails. 2. Failed to convert a list of field names into corresponding field IDs. 3. System ran out of memory. 4. trapd has detected that the connection to netmon has closed, but netmon did not send a close_event. 5. trapd has detected that the connection to snmpCollect has closed, but snmpCollect did not send a close_event. 6. trapd has detected that the connection to ovtopmd has closed, but ovtopmd did not send a close_event. 7. trapd is closing the connection to *applicationName* because trapd has reached the maximum number of outstanding events that it will queue up. 8. trapd is closing the connection to the generic application, because trapd has queue up the maximum number of events for the application.
58916864 NUP_EV Node Up	Descr field	Node Up
	Data field	Contains the following: • Time stamp • Object ID
	Condition	All detected interfaces for the node are up.
58916865 NDWN_EV Node Down	Descr field	Node Down
	Data field	Contains the following: • Time stamp • Object ID
	Condition	All detected interfaces for the node are down.

Table 19 (Page 12 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
58916866 IUP_EV Interface Up	Descr field	Interface *interfaceLabel* up.
	Data field	IP address, the time stamp of the event, the node and interface ids.
	Condition	An interface that was previously down has responded to a ping.
58916867 IDWN_EV Interface Down	Descr field	Interface *interfaceLabel* Down.
	Data field	IP address, the time stamp of the event, the node and interface ids.
	Condition	An interface that was previously up is not responding to a ping.
58916868 SC_EV Segment Critical	Descr field	Segment *segmentId* Down
	Data field	Contains the following: • Time stamp • Segment object ID
	Condition	All nodes within a segment are critical.
58916869 NC_EV Network Critical	Descr field	Network *netName* Down
	Data field	Time stamp of the event and network objid
	Condition	All segments for the network are critical.

58916871
SNMP_EV
SNMP Status Event - Not generated
	This is used for Configurable Status. MIB variables should contain:
	Varbind-1: 14
	Varbind-2: *Selection name of object*
	Varbind-3: "Object status is"
	Varbind-4: One of the words:
	`Unknown Normal Marginal Critical`
	`Up Down User1 User2`	

58916964 NV6KUP_EV NetView for AIX Up	Descr field	NetView for AIX Version 4 *nodeName* Up
	Data field	Contains the following: Time stamp Node name
	Condition	NetView for AIX Version 4 is running on this node.
58916965 NV6KDN_EV NetView for AIX Down	Descr field	NetView for AIX Version 4 *nodeName* Down
	Data field	Contains the following: Time stamp Node name
	Condition	NetView for AIX Version 4 is no longer running on this node.

Table 19 (Page 13 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
58982400 LLAC_EV Link Level Address Changed	Descr field	Link Address For *interfaceLabel* Changed to *physicalAddress*
	Data field	IP address, the time stamp of the event, node and interface objids
	Condition	The product has detected that interfaceLabel has a new physical Address.
58982401 MLLA_EV Mismatch of Link Level Address	Descr field	*nodeName1* reported different Link Address than obtained from *nodeName2* by SNMP
	Data field	nodeName of the node that reported different address
	Condition	NetView for AIX has discovered a discrepancy in the LLA being reported by 2 different nodes. nodeName1's ifTable is being interrogated for LLA vs. the information in nodeName2's ARP cache. This can be diagnosed by using `rnetstat -I nodeName1` to determine the interfaces and LLA for nodeName1, and then issue `rnetstat -A nodeName2 | grep nodeName1` to determine the nodeName2 ARP entry for nodeName1. It is possible that the information stored in the topology database for nodeName1 is out of date. If so, performing a demand poll on nodeName1 should resolve the situation. The command: `ovtopodump -L | grep nodeName1` could also be performed to check the LLA that NetView for AIX currently has associated with the interfaces for nodeName1. Some vendors make wide assumptions about documenting information in the ARP cache, thus there are many reasons why a mismatch may occur, but the network configuration is still valid. Netmon supports a flag to disable this check completely. To disable, select the option **ignore** under the netmon SMIT configure daemons options for Ring bit-swapping storage flag.

Table 19 (Page 14 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
58982402 ULLA_EV Undetermined Link Level Address	Descr field	One of the following description fields is possible with this trap: • If there are two nodes that have reported different LLA address then the message displayed is "*nodeName* reports a different Link Address for this node from that reported by *secondNodeName*" • If the same node reports a different LLA address then what is reported earlier then the message displayed is "*nodeName* reports a different Link Address for this node than it reported earlier."
	Data field	nodeName, secondNodeName
		nodeName The name of the node which has reported an LLA address which is different from the one which this (reporting) node got earlier either from the same node (nodeName = secondNodeName) or from a different node (nodeName != secondNodeName)
		secondNodeName Name of the node which had earlier reported LLA address.
	Condition	NodeName gives LLA address for interface which is different than we obtained in the past from the IP Address. If this message keeps repeating, it indicates that the LLA for interface keeps flopping; this usually means there is more than one interface with the same IP address. If it happens once or twice and settles down, it usually means that a LAN card has changed. There might be valid network reasons why the ULLA_EV is being displayed. The same SMIT option described above to disable the MLLA_EV would disable the ULLA_EV.
58982403 OIC_EV Object Identifier Change	Descr field	Object Identifier changed to *newObjectId*
	Data field	Time stamp, node object ID, new sysObject ID
	Condition	A new sysObjectID was detected
58982404 SDC_EV System Description Change	Descr field	System Description Changed to *sysDescr*
	Data field	Contains the following: • Time stamp • Node object ID • New sysDescr
	Condition	A new sysDescr was detected

Table 19 (Page 15 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
58982405 SNC_EV System Name Change	Descr field	System Name changed
	Data field	Contains the following: • Time stamp • Node object ID • New sysName
	Condition	A new sysName was detected
58982406 SMC_EV Subnet Mask Change	Descr field	Network Mask for *interfaceLabel* changed to *inetAddr*
	Data field	IP address, the time stamp of the event, node id, interface id and interface type.
	Condition	A new subnet mask for an interface was detected
58982407 FSC_EV Forwarding Status Change	Descr field	It can be one of the following fields: • Forwarding status changed -- Now a Gateway • Forwarding status changed -- Now a host • Forwarding status changed -- Now Unknown
	Data field	Contains the following: • Time stamp • Node object ID • 1 if a gateway and 2 if a host
	Condition	A change was detected in a node's ipForwarding status.
58982408 FTH_EV Forwarding to a host	Descr field	Incorrect Routing to node *nodeName*
	Data field	nodeName
	Condition	The node for the event contains a route in its routing table to nodeName, yet our topology database entry for nodeName indicates that its ipForwarding status is not a gateway. This could be corrected via a demand poll to nodeName which would update its ipForwarding status in the database. This event could also be generated if nodeName is not allowing SNMP communication to the manager. In this case, the manager cannot determine the ipForwarding status for nodeName.
58982410 SCC_EV System Contact Change	Descr field	System Contact Changed to *contactName*
	Data field	Contains the following: • Time stamp • Node object ID • New system contact
	Condition	Detected a new sysContact for the node.

Table 19 (Page 16 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
58982411 SLC_EV System Location Change	Descr field	System Location Changed to *nodeLocation*
	Data field	Contains the following: • Time stamp • Node object ID • New system location
	Condition	Detected a new sysLocation for the node.
58982412 ITC_EV Interface Type Change	Descr field:	Interface Type for *interfaceLabel* changed to *typeName*
	Data field	IP address, time stamp, node id, interface ids and interface type.
	Condition	Detected a new interface type for an interface.
58982413 IDC_EV Interface Description Change	Descr field	Interface Descriptor for *interfaceLabel* changed to *interfaceName*
	Data field	IP address, time stamp, node id, interface ids and interface description.
	Condition	Detected a new interface description for an interface.
58982414 BSM_EV Bad Subnet Mask	Descr field	Inconsistent subnet mask *subnetMask* on interface *inetAddr*
	Data field	Bad interface address and bad subnet mask
	Condition	The subnet mask obtained for an IP address does not match the assumed subnet mask for the subnet in which that interface resides.
59047936 AA_EV Application Alert	Descr field	*alertapp: classStr: alertMsg* **alertApp** The application name which has sent the alert **classStr** Can take one of the following values: `INFORMATION WARNING ERROR DISASTER` **alertMsg** The Alert message sent by the application
	Data field	NULL
	Condition	When an application makes use of the OVwAlertMsg API to send a message.
59179056 APUP_EV Application Up event	Descr field	*applicationName* connected to trapd
	Data field	NULL
	Condition	When an application gets connected to trapd via its internal socket API.
59179057 APDN_EV Application Down Event	Descr field	*applicationName* disconnecting from trapd
	Data field	NULL
	Condition	When an application disconnects from trapd, this event is generated.

Table 19 (Page 17 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
59179068 TATM_EV Tralertd Change Tracemask Event	Descr field	Tralertd trace mask change request
	Data field	tracemask
	Condition	The tralertd daemon uses this for on-the-fly changing of its tracemask.
59179070 NMCR_EV Netmon Change Retry Count Event	Descr field	Netmon retry count changed
	Data field	retry count value
	Condition	Generated when the netmon option to change the retry count is issued. This event is obsolete with the retries allowed via the ovsnmp.conf file.
70000030 NVOT001 Vertex Created	Descr field	Can be one of the following values: • namebinding(int) • protocol(int) • name(string) • detailsld(string)
	Condition	Creation of a vertex in gtmd
70000031 NVOT002 Vertex Deleted	Descr field	Can be one of the following values: • namebinding(int) • protocol(int) • name(string)
	Condition	Deletion of a vertex in gtmd
70000032 NVOT003 Vertex Status Changed	Descr field	Can be one of the following values: • namebinding(int) • protocol(int) • name(string)
	Condition	Status change of a vertex in gtmd
70000033 NVOT004 Graph Created	Descr field	Can be one of the following values: • namebinding(int) • protocol(int) • name(string) • detailsld(string)
	Condition	Creation of a graph or box in gtmd
70000034 NVOT005 Graph Deleted	Descr field	Can be one of the following values: • namebinding(int) • protocol(int) • name(string)
	Condition	Deletion of a graph or box in gtmd

Table 19 (Page 18 of 18). NetView for AIX Internal Traps

Number, Name, and Description	Fields and Condition	
70000035 NVOT006 Arc Created	Descr field	Can be one of the following values: • namebinding(int) • Aprotocol(int) • Aname(string) • Zprotocol(int/str) • Zname(string) • arcindexld(int) • detailsld(string)
	Condition	Creation of an arc in gtmd
70000036 NVOT007 Arc Deleted	Descr field	Can be one of the following values: • namebinding(int) • Aprotocol(int) • Aname(string) • Zprotocol(int/str) • Zname(string) • arcindexld(int)
	Condition	Deletion of an arc in gtmd
70000037 NVOT008 Arc Status Changed	Descr field	Can be one of the following values: • namebinding(int) • Aprotocol(int) • Aname(string) • Zprotocol(int/str) • Zname(string) • arcindexld(int)
	Condition	Status change of an arc in gtmd

Glossary, Bibliography, and Index

Glossary

This glossary includes terms and definitions from:

- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 11 West 42nd Street, New York, New York 10036. Definitions are identified by the symbol (A) after the definition.
- The ANSI/EIA Standard—440-A, *Fiber Optic Terminology*. Copies may be purchased from the Electronic Industries Association, 2001 Pennsylvania Avenue, N.W., Washington, DC 20006. Definitions are identified by the symbol (E) 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 Network Working Group Request for Comments: 1208.
- The *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994.
- The *Object-Oriented Interface Design: IBM Common User Access Guidelines*, Carmel, Indiana: Que, 1992.

The following cross-references are used in this glossary:

Contrast with: This refers to a term that has an opposed or substantively different meaning.

Synonym for: This indicates that the term has the same meaning as a preferred term, which is defined in its proper place in the glossary.

Synonymous with: This is a backward reference from a defined term to all other terms that have the same meaning.

See: This refers the reader to multiple-word terms that have the same last word.

See also: This refers the reader to terms that have a related, but not synonymous, 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.

A

abstract syntax. A data specification that includes all distinctions that are needed in data transmissions, but that omits (abstracts) other details such as those that depend on specific computer architectures. See also *abstract syntax notation 1 (ASN.1)* and *basic encoding rules (BER)*.

abstract syntax notation 1 (ASN.1). The Open Systems Interconnection (OSI) method for abstract syntax specified in ISO 8824. See also *basic encoding rules (BER)*.

ACB name. (1) The label of an ACB macroinstruction used to name the ACB. (2) A name specified either on the VTAM APPL definition statement or on the VTAM application program's ACB macroinstruction. Contrast with *network name*.

action. (1) An operation on a managed object, the semantics of which are defined as part of the managed object class definition. (2) In the AIX operating system, a defined task that an application performs. An action modifies the properties of an object or manipulates the object in some way.

active. (1) The state of a resource when it has been activated and is operational. (2) In the AIX operating system, pertaining to the window pane in which the text cursor is currently positioned. (3) Contrast with *inactive* and *inoperative*.

adapter. A part that electrically or physically connects a device to a computer or to another device.

address mask. For internet subnetworking, a 32-bit mask used to identify the subnetwork address bits in the host portion of an IP address. Synonymous with *subnet mask* and *subnetwork mask*.

Address Resolution Protocol (ARP). (1) In the Internet suite of protocols, the protocol that dynamically

maps an IP address to an address used by a supporting metropolitan or local area network such as Ethernet or token-ring. (2) See also *Reverse Address Resolution Protocol (RARP)*.

Administrative Domain. A collection of hosts and routers, and the interconnecting networks, managed by a single administrative authority.

agent. (1) In systems management, a user that, for a particular interaction, has assumed an agent role. (2) An entity that represents one or more managed objects by (a) emitting notifications regarding the objects and (b) handling requests from managers for management operations to modify or query the objects. (3) A system that assumes an agent role.

aggregate. In programming languages, a structured collection of data objects that form a data type. (I)

AIX. Advanced Interactive Executive.

AIX NetView Service Point. See *NetView for AIX Service Point*.

AIX NetView/6000. See *NetView for AIX*.

AIX operating system. IBM's implementation of the UNIX operating system. The RISC System/6000 system, among others, runs the AIX operating system.

AIX SystemView NetView/6000. See *NetView for AIX*.

alert. (1) A message sent to a management services focal point in a network to identify a problem or an impending problem. (2) In SNA management services (SNA/MS), a high priority event that warrants immediate attention.

API. Application programming interface.

application plane. In NetView for AIX, the submap layer on which symbols of objects that are managed by at least one network or systems management application program are displayed. Symbols on the application plane are displayed without shading, which makes them appear directly against the background plane. See also *user plane*.

application programming interface (API). The set of programming language constructs or statements that can be coded in an application program to obtain the specific functions and services provided by an underlying operating system or service program.

application registration file. A file created to integrate an application program into NetView for AIX by defining (a) the application program's position in the menu structure for NetView for AIX, (b) where help information is found, (c) the number and types of parameters allowed, (d) the command used to start the application program, and (e) other characteristics of the application program.

Apply. A push button that carries out the selected choices in a window without closing the window.

arc. In graphs, a curve or line segment that links two vertices.

ARP. Address Resolution Protocol.

ASCII (American National Standard Code for Information Interchange). The standard code, using a coded character set consisting of 7-bit coded characters (8 bits including parity check), that is used for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters. (A)

ASN.1. Abstract syntax notation 1.

attribute. (1) A characteristic that identifies and describes a managed object. The characteristic can be determined, and possibly changed, through operations on the managed object. (2) Information within a managed object that is visible at the object boundary. An attribute has a type, which indicates the range of information given by the attribute, and a value, which is within that range. (3) Variable data that is logically a part of an object and that represents a property of the object. For example, a serial number is an attribute of an equipment object.

authentication. (1) In computer security, verification of the identity of a user or the user's eligibility to access an object. (2) In computer security, verification that a message has not been altered or corrupted. (3) In computer security, a process used to verify the user of an information system or protected resources.

authentication failure. In the Simple Network Management Protocol (SNMP), a trap that may be generated by an authentication entity when a requesting client is not a member of the SNMP community.

authorization. (1) In computer security, the right granted to a user to communicate with or make use of a computer system. (T) (2) An access right. (3) The

process of granting a user either complete or restricted access to an object, resource, or function.

B

background picture. The diagram or image that is displayed behind other symbols to show their context or relations.

background plane. In NetView for AIX, the lowest submap layer. The background plane provides the background against which symbols are displayed. A background picture can be placed in the background plane to provide a context for viewing symbols. See also *application plane* and *user plane*.

basic encoding rules (BER). The rules specified in ISO 8825 for encoding data units described in abstract syntax notation 1 (ASN.1). The rules specify the encoding technique, not the abstract syntax.

behavior. (1) Ideally, a collection of assertions that describe the allowed states that a managed object can assume. An assertion can be a precondition, a postcondition, or an invariant. In practice, the behavior is often an informal description of the semantics of attributes, operations, and notifications. (2) The way in which managed objects, name bindings, attributes, notifications, and operations interact with the actual resources that they model and with each other.

BER. Basic encoding rules.

bind. To relate an identifier to another object in a program; for example, to relate an identifier to a value, an address or another identifier, or to associate formal parameters and actual parameters. (T)

bridge. (1) A functional unit that interconnects two local area networks that use the same logical link control protocol but may use different medium access control protocols. (T) (2) A functional unit that interconnects multiple LANs (locally or remotely) that use the same logical link control protocol but that can use different medium access control protocols. A bridge forwards a frame to another bridge based on the medium access control (MAC) address. (3) In the connection of local loops, channels, or rings, the equipment and techniques used to match circuits and to facilitate accurate data transmission. (4) Contrast with *gateway* and *router*.

browse. To look at records in a file.

buffer. (1) To allocate and schedule the use of buffers. (A) (2) A portion of storage used to hold input or output data temporarily.

bus. (1) A facility for transferring data between several devices located between two end points, only one device being able to transmit at a given moment. (T) (2) A computer configuration in which processors are interconnected in series.

button. (1) A mechanism on a pointing device, such as a mouse, used to request or initiate an action or a process. (2) A graphical device that identifies a choice. (3) A graphical mechanism that, when selected, performs a visible action. For example, when a user clicks on a list button, a list of choices appears. (4) See *mouse button*, *push button*, *radio button*, and *spin button*.

C

cache. (1) A special-purpose buffer storage, smaller and faster than main storage, used to hold a copy of instructions and data obtained from main storage and likely to be needed next by the processor. (T) (2) A buffer storage that contains frequently accessed instructions and data; it is used to reduce access time. (3) An optional part of the directory database in network nodes where frequently used directory information may be stored to speed directory searches. (4) To place, hide, or store in a cache.

Cancel. A push button that removes a window without applying any changes made in that window.

card. In NetView for AIX, see *event card*.

child process. In the AIX and OS/2 operating systems, a process, started by a parent process, that shares the resources of the parent process. See also *fork*.

class. (1) In object-oriented design or programming, a group of objects that share a common definition and that therefore share common properties, operations, and behavior. Members of the group are called instances of the class. (2) In the AIX operating system, pertaining to the I/O characteristics of a device. System devices are classified as block or character devices.

click. To press and release a button on a pointing device without moving the pointer off of the object or choice.

client. (1) A functional unit that receives shared services from a server. (T) (2) A user. (3) In an AIX distributed file system environment, a system that is dependent on a server to provide it with programs or access to programs.

client/server. In communications, the model of interaction in distributed data processing in which a program at one site sends a request to a program at another site and awaits a response. The requesting program is called a client; the answering program is called a server.

Close. A choice that removes a window and all of the windows associated with it from the workplace. For example, if a user is performing a task in a window and a message appears, or the user asks for help, both the message and the help windows disappear when the user closes the original window.

CMIP. Common Management Information Protocol.

CMIS. Common Management Information Service.

CMOT. Common Management Information Protocol over TCP/IP.

command. A request from a terminal for the performance of an operation or the execution of a particular program.

Common Management Information Protocol (CMIP). The OSI standard protocol defined in ISO/IEC 9596-1 for the interaction between managers and agents that use the Common Management Information Service Element (CMISE).

Common Management Information Protocol over TCP/IP (CMOT). An Internet Engineering Task Force (IETF) specification for the use of CMIP over a TCP/IP protocol stack.

Common Management Information Service (CMIS). The set of services provided by the Common Management Information Service Element.

communications infrastructure. In the AIX operating system, a framework of communication that consists of a postmaster, an object registration service, a startup file, communication protocols, and application programming interfaces.

community. In the Simple Network Management Protocol (SNMP), an administrative relationship between entities.

community name. In the Simple Network Management Protocol (SNMP), a string of octets identifying a community.

component. Hardware or software that is part of a functional unit.

configuration. (1) The manner in which the hardware and software of an information processing system are organized and interconnected. (T) (2) The devices and programs that make up a system, subsystem, or network.

configuration file. A file that specifies the characteristics of a system device or network.

connection. (1) In data communication, an association established between functional units for conveying information. (I) (A) (2) In Open Systems Interconnection architecture, an association established by a given layer between two or more entities of the next higher layer for the purpose of data transfer. (T) (3) In TCP/IP, the path between two protocol applications that provides reliable data stream delivery service. In the Internet, a connection extends from a TCP application on one system to a TCP application on another system. (4) In system communications, a line over which data can be passed between two systems or between a system and a device. (5) Synonym for *physical connection*.

connector class. In NetView for AIX, an object class used for objects that connect different parts of the network and that route or switch traffic between these parts. This class includes gateways, repeaters (including multiport repeaters), and bridges. Contrast with *network class*.

container. A visual user-interface component whose specific purpose is to hold objects.

control desk. In NetView for AIX, a component of the graphical user interface (GUI) that enables the network operator to group application program instances together.

Copy. A choice that places a copy of a selected object onto the clipboard.

cron table. In the AIX operating system, a table used to schedule application programs and processes.

Note: “Cron” is an abbreviation for “chronological.”

Cut. A choice that moves a selected object and places it onto the clipboard. The space it occupied is usually filled by the remaining object or objects in the window.

D

daemon. A program that runs unattended to perform a standard service. Some daemons are triggered automatically to perform their task; others operate periodically.

data. A representation of facts or instructions in a form suitable for communication, interpretation, or processing by human or automatic means. Data include constants, variables, arrays, and character strings.

Note: Programmers make a distinction between instructions and the data they operate on; however, in the usual sense of the word, data includes programs and program instructions.

data circuit-terminating equipment (DCE). In a data station, the equipment that provides the signal conversion and coding between the data terminal equipment (DTE) and the line. (I)

Notes:

1. The DCE may be separate equipment or an integral part of the DTE or of the intermediate equipment.
2. A DCE may perform other functions that are usually performed at the network end of the line.

data set. Synonym for *file*.

datagram. In TCP/IP, the basic unit of information passed across the Internet environment. A datagram contains a source and destination address along with the data. An Internet Protocol (IP) datagram consists of an IP header followed by the transport layer data.

DCE. (1) Data circuit-terminating equipment. (2) Distributed Computing Environment.

default. Pertaining to an attribute, condition, value, or option that is assumed when none is explicitly specified. (I)

Delete. A choice that removes a selected object. The space it occupied is usually filled by the remaining object or objects in the window.

demand poll. In NetView for AIX, a polling operation initiated by the user.

destination. Any point or location, such as a node, station, or a particular terminal, to which information is to be sent.

device. A mechanical, electrical, or electronic contrivance with a specific purpose.

dialog box. In OSF/Motif, a collection of data fields and buttons for setting controls, selecting from lists, choosing from mutually exclusive options, entering data, and presenting the user with messages.

disable. To make nonfunctional.

discovery. In data communication, the automatic detection of network topology changes (for example, new and deleted nodes or new and deleted interfaces).

display. (1) A visual presentation of data. (I) (A) (2) To present data visually. (I) (A) (3) Deprecated term for *panel*.

display panel. In computer graphics, a predefined display image that defines the locations and characteristics of display fields on a display surface.

Distributed Computing Environment (DCE). The Open Software Foundation (OSF) specification (or a product derived from this specification) that assists in networking. DCE provides such functions as authentication, directory service (DS), and remote procedure call (RPC).

DNS. Domain Name System.

domain. (1) That part of a computer network in which the data processing resources are under common control. (T) (2) In SNA, see *end node domain*, *network node domain*, and *system services control point (SSCP) domain*. (3) In Open Systems Interconnection (OSI), a part of a distributed system or a set of managed objects to which a common policy applies. (4) In a database, all the possible values of an attribute or a data element. (5) See *Administrative Domain* and *domain name*.

domain name. In the Internet suite of protocols, a name of a host system. A domain name consists of a sequence of subnames separated by a delimiter character. For example, if the fully qualified domain name (FQDN) of a host system is `ralvm7.vnet.ibm.com`, each of the following is a domain name:

- `ralvm7.vnet.ibm.com`
- `vnet.ibm.com`
- `ibm.com`

Domain Name System (DNS). In the Internet suite of protocols, the distributed database system used to map domain names to IP addresses.

domain operator. In a multiple-domain network, the person or program that controls operation of resources controlled by one system services control point (SSCP). See also *network operator*.

DOS. Disk Operating System. See *IBM Disk Operating System*.

double-click. To press and release a button on a pointing device twice while a pointer is within the limits that the user has specified for the operating environment.

drag. To use a pointing device to move an object. For example, a user can drag a window border to make the window larger.

drag and drop. To directly manipulate an object by moving it and placing it somewhere else using a pointing device.

dump. (1) To record, at a particular instant, the contents of all or part of one storage device in another storage device. Dumping is usually for the purpose of debugging. (T) (2) Data that has been dumped. (T) (3) To copy data in a readable format from main or auxiliary storage onto an external medium such as tape, diskette, or printer.

dynamic. (1) In programming languages, pertaining to properties that can only be established during the execution of a program; for example, the length of a variable-length data object is dynamic. (I) (2) Pertaining to an operation that occurs at the time it is needed rather than at a predetermined or fixed time. (3) Contrast with *static*.

E

e-mail. Electronic mail.

echo. (1) In computer graphics, the immediate notification of the current values provided by an input device to the operator at the display console. (I) (A) (2) In data communication, a reflected signal on a communications channel. For example, on a communications terminal, each signal is displayed twice, once when entered at the local terminal and again when returned over the communications link. This allows the signals to be checked for accuracy.

EFD. Event forwarding discriminator.

electronic mail (e-mail). (1) Correspondence in the form of messages transmitted between user terminals over a computer network. (T) (2) The generation, transmission, and display of correspondence and documents by electronic means. (A)

enable. To make functional.

end node domain. An end node control point, its attached links, and its local LUs.

end-user interface (EUI). In NetView for AIX, synonym for *graphical user interface (GUI)*.

entity. Any concrete or abstract thing of interest, including associations among things; for example, a person, object, event, or process that is of interest in the context under consideration, and about which data may be stored in a database. (T)

error. A discrepancy between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition. (I) (A)

Ethernet. A 10-Mbps baseband local area network that allows multiple stations to access the transmission medium at will without prior coordination, avoids contention by using carrier sense and deference, and resolves contention by using collision detection and delayed retransmission. Ethernet uses carrier sense multiple access with collision detection (CSMA/CD).

EUI. End-user interface.

event. (1) An occurrence of significance to a task; for example, an SNMP trap, the opening of a window or a submap, or the completion of an asynchronous operation. (2) In the NetView and NETCENTER programs, a record indicating irregularities of operation in physical elements of a network. (3) See also *event report*.

event card. In NetView for AIX, a graphical representation, resembling a card, of the information contained in an event sent by an agent to a manager reflecting a change in the status of one of the agent's managed nodes.

event filter. In NetView for AIX, a logical expression of criteria that determine which events are forwarded to the application program that registers the event filter with the event sieve agent. A filter is referred to as “simple” or “compound” depending on how it is handled by the filter editor.

event forwarding discriminator (EFD). A managed object that describes and controls the criteria used to select which event reports are sent and to whom they are sent.

event management services (EMS). In NetView for AIX, a centralized method of generating, receiving, routing, and logging network events.

event report. The unsolicited report that an event has occurred. When a managed object emits a notification, the agent uses one or more event forwarding discriminators (EFDs) to find the destinations to which the report is sent.

event sieve. In NetView for AIX, an object that is managed by the "ovesmd" daemon, which is the event sieve agent. The event sieve agent stores information about the event sieve object in a database and reads that information when the agent is started. See also *event filter* and *event forwarding discriminator (EFD)*.

exec. (1) In the AIX operating system, to overlay the current process with another executable program. (2) See also *fork*.

executable symbol. In NetView for AIX, a symbol defined such that double-clicking on it causes an application program to perform an action on a set of target objects. Contrast with *explodable symbol*.

explodable symbol. In NetView for AIX, a symbol defined such that double-clicking on it or dragging and dropping it displays the child submap of the parent object that the symbol represents. Contrast with *executable symbol*.

F

FDDI. Fiber Distributed Data Interface.

feature. A part of an IBM product that may be ordered separately by the customer.

Fiber Distributed Data Interface (FDDI). An American National Standards Institute (ANSI) standard for a 100-megabit-per-second LAN using optical fiber cables.

field. (1) An identifiable area in a window. Examples of fields are: an entry field, into which a user can type or place text, and a field of radio button choices, from which a user can select one choice. (2) In NetView for AIX, the building block of which objects are composed. A field is characterized by a field name, a data type (integer, Boolean, character string, or enumerated value), and a set of flags that describe how the field is treated by NetView for AIX. A field can contain data only when it is associated with an object.

field registration file. In NetView for AIX, a file used to define fields for use in the object database.

file. A named set of records stored or processed as a unit. (T) Synonymous with *data set*.

filter. (1) A device or program that separates data, signals, or material in accordance with specified criteria. (A) (2) In the NetView program, a function that limits the data that is to be recorded on the database and displayed at the terminal. (3) In the AIX operating system, a command that reads standard input data, modifies the data, and sends it to the display screen. (4) See also *recording filter* and *viewing filter*.

filter editor. In NetView for AIX, a part of the graphical user interface (GUI) that enables the user to define, modify, and delete filtering rules for use by application programs.

flag. (1) To mark an information item for selection for further processing. (T) (2) A character that signals the occurrence of some condition, such as the end of a word. (A)

flat file. (1) A one-dimensional or two-dimensional array: a list or table of items. (2) In a relational database, synonym for *relation*. (3) A file that has no hierarchical structure.

focal point (FP). In the NetView program, 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.

font. A family of characters of a given size and style; for example, 9-point Helvetica. (T)

fork. In the AIX operating system, to create and start a child process.

FP. Focal point.

FQDN. Fully qualified domain name.

fully qualified domain name (FQDN). In the Internet suite of protocols, the name of a host system that includes all of the subnames of the domain name. An example of a fully qualified domain name is `ralvm7.vnet.ibm.com`. See also *host name*.

G

gateway. (1) A functional unit that interconnects two computer networks with different network architectures. A gateway connects networks or systems of different architectures. A bridge interconnects networks or systems with the same or similar architectures. (T) (2) In the AIX operating system, an entity that operates above the link layer and translates, when required, the interface and protocol used by one network into those used by another distinct network. (3) In TCP/IP, synonym for *router*.

general topology manager (GTM). In NetView for AIX, the component that accepts information about resources that are accessed through protocols other than the Internet Protocol (IP), stores this information in a database, and displays it to the user.

generic alert. A product-independent method of encoding alert data by means of both (a) code points indexing short units of stored text and (b) textual data.

GIF. Graphical interchange format.

graph. A set of vertices and the set of arcs that link pairs of those vertices.

graphical interchange format (GIF). In NetView for AIX, the format used for the background pictures of a network topology map.

graphical user interface (GUI). (1) A type of computer interface consisting of a visual metaphor of a real-world scene, often of a desktop. Within that scene are icons, representing actual objects, that the user can access and manipulate with a pointing device. (2) In NetView for AIX, the integrating interface application program that provides the means for displaying submaps and for integrating network application programs. The graphical user interface is a single, consistent interface that enables the user to interact with multiple application programs. Synonymous with *end-user interface (EUI)*.

GTM. General topology manager.

GUI. Graphical user interface.

H

hardcopy. (1) A permanent copy of a display image generated on an output device such as a printer or plotter, and which can be carried away. (T) (2) A printed copy of machine output in a visually readable form; for example, printed reports, listings, documents, and summaries. (3) Contrast with *softcopy*.

Help. A choice that gives a user access to helpful information about objects, choices, tasks, and products. A Help choice can appear on a menu bar or as a push button.

help panel. Information displayed by a system in response to a help request from a user.

hierarchy. The resource types, display types, and data types that make up the organization, or levels, in a network.

highlighting. Emphasizing a display element or segment by modifying its visual attributes. (I) (A)

home submap. In NetView for AIX, the first submap that appears when a map is opened. Each map has a home submap. When new maps are created, the home submap is the root submap.

host. (1) In the Internet suite of protocols, an end system. The end system can be any workstation; it does not have to be a mainframe. (2) See *host processor*.

host name. In the Internet suite of protocols, the name given to a machine. Sometimes, “host name” is used to mean *fully qualified domain name (FQDN)*; other times, it is used to mean the most specific subname of a fully qualified domain name. For example, if `ralvm7.vnet.ibm.com` is the fully qualified domain name, either of the following may be considered the host name:

- `ralvm7.vnet.ibm.com`
- `ralvm7`

host processor. (1) A processor that controls all or part of a user application network. (T) (2) In a network, the processing unit in which the data communication access method resides.

I

IBM Disk Operating System (DOS). A disk operating system based on MS-DOS that operates with all IBM personal computers.

ICMP. Internet Control Message Protocol.

icon. (1) A graphic symbol, displayed on a screen, that a user can point to with a device such as a mouse in order to select a particular function or software application. (T) (2) A graphical representation of an object, consisting of an image, image background, and a label.

ID. (1) Identifier. (2) Identification.

IEEE. Institute of Electrical and Electronics Engineers.

inactive. (1) Not operational. (2) Pertaining to a node or device not connected or not available for connection to another node or device. (3) In the AIX operating system, pertaining to a window that does not have an input focus. (4) Contrast with *active*. (5) See also *inoperative*.

inoperative. (1) The condition of a resource that has been active but is not currently active. A resource may be inoperative for reasons such as the following: (a) it may have failed, (b) it may have received an INOP request, or (c) it may be suspended while a reactivate command is being processed. (2) See also *inactive*.

instance. In the AIX operating system, a concrete realization of an abstract object class. An instance of a widget or a gadget is a specific data structure that contains detailed appearance and behavioral information that is used to generate a specific graphical object on-screen at run time.

Institute of Electrical and Electronics Engineers (IEEE). A professional society accredited by the American National Standards Institute (ANSI) to issue standards for the electronics industry.

interface. A shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics, as appropriate. The concept includes the specification of the connection of two devices having different functions. (T)

International Organization for Standardization (ISO). An organization of national standards bodies from various countries established to promote development of standards to facilitate international exchange of goods and services, and develop cooperation in intellectual, scientific, technological, and economic activity.

internet. A collection of networks interconnected by a set of routers that allow them to function as a single, large network. See also *Internet*.

Internet. The internet administered by the Internet Architecture Board (IAB), consisting of large national backbone networks and many regional and campus networks all over the world. The Internet uses the Internet suite of protocols.

Internet Control Message Protocol (ICMP). The protocol used to handle errors and control messages in the Internet Protocol (IP) layer. Reports of problems and incorrect datagram destinations are returned to the original datagram source. ICMP is part of the Internet Protocol.

internet object. In NetView for AIX, a node or a network that can be addressed by the Internet Protocol (IP).

Internet Protocol (IP). A connectionless protocol that routes data through a network or interconnected networks. IP acts as an intermediary between the higher protocol layers and the physical network. However, this protocol does not provide error recovery and flow control and does not guarantee the reliability of the physical network.

Internetwork Packet Exchange (IPX). The network protocol used to connect Novell's servers, or any workstation or router that implements IPX, with other workstations. Although similar to the Internet Protocol (IP), IPX uses different packet formats and terminology.

interprocess communication (IPC). (1) In the AIX operating system, the process by which programs communicate data to each other and synchronize their activities. Semaphores, signals, and internal message queues are common methods of interprocess communication. (2) In the Enhanced X-Windows Toolkit, a communication path. See also *client*.

IP. Internet Protocol.

IP address. The 32-bit address defined by the Internet Protocol, standard 5, Request for Comment (RFC) 791. It is usually represented in dotted decimal notation.

IPC. Interprocess communication.

IPX. Internetwork Packet Exchange.

ISO. International Organization for Standardization.

K

keyword. (1) In programming languages, a lexical unit that, in certain contexts, characterizes some language construct; for example, in some contexts, IF characterizes an if-statement. A keyword normally has the form of an identifier. (I) (2) One of the predefined words of an artificial language. (A) (3) A name or symbol that identifies a parameter. (4) The part of a command operand that consists of a specific character string (such as `DSNAME=`).

L

LAN. Local area network.

layout. See *layout algorithm*.

layout algorithm. A method of arranging displayed or printed data.

link. The combination of the link connection (the transmission medium) and two link stations, one at each end of the link connection. A link connection can be shared among multiple links in a multipoint or token-ring configuration.

list button. A button labeled with an underlined down-arrow that presents a list of valid objects or choices that can be selected for that field.

local. (1) Pertaining to a device accessed directly without use of a telecommunication line. (2) Contrast with *remote*.

local area network (LAN). (1) A computer network located on a user's premises within a limited geographical area. Communication within a local area network is not subject to external regulations; however, communication across the LAN boundary may be subject to some form of regulation. (T) (2) A network in which a set of devices are connected to one another for communication and that can be connected to a larger network. (3) See also *Ethernet* and *token ring*. (4) Contrast with *metropolitan area network (MAN)* and *wide area network (WAN)*.

local host. (1) In TCP/IP, the host on the network at which a particular operator is working. (2) In an internet, the host to which a user's terminal is connected without using the internet.

local registration file (LRF). In NetView for AIX, a file that provides information about an agent or daemon, such as the name, the location of the executable code, the names of processes dependent on the agent or daemon, and details about the objects that an agent manages.

LRF. Local registration file.

M

mainframe. A computer, usually in a computer center, with extensive capabilities and resources to which other computers may be connected so that they can share facilities. (T)

MAN. Metropolitan area network.

managed node. In Internet communications, a workstation, server, or router that contains a network management agent. In the Internet Protocol (IP), the managed node usually contains a Simple Network Management Protocol (SNMP) agent.

managed object. (1) A component of a system that can be managed by a management application. (2) The systems management view of a resource that can be managed through the use of systems management protocols.

Management Information Base (MIB). (1) A collection of objects that can be accessed by means of a network management protocol. (2) A definition for management information that specifies the information available from a host or gateway and the operations allowed. (3) In OSI, the conceptual repository of management information within an open system.

management region. In NetView for AIX, the set of managed objects on a particular map that defines the extent of the network that is being actively managed. The management region may vary across maps.

management services (MS). (1) One of the types of network services in control points (CPs) and physical units (PUs). Management services are the services provided to assist in the management of SNA networks, such as problem management, performance and accounting management, configuration management, and change management. (2) Services that assist in the management of systems and networks in areas such as problem management, performance management,

business management, operations management, configuration management, and change management.

manager. (1) In systems management, a user that, for a particular interaction, has assumed a manager role. (2) An entity that monitors or controls one or more managed objects by (a) receiving notifications regarding the objects and (b) requesting management operations to modify or query the objects. (3) A system that assumes a manager role.

map. In NetView for AIX, a database represented by a set of related submaps that provide a graphical and hierarchical presentation of a network and its systems.

medium. A physical material in or on which data may be represented.

menu. (1) A list of options displayed to the user by a data processing system, from which the user can select an action to be initiated. (T) (2) A list of choices that can be applied to an object. A menu can contain choices that are not available for selection in certain contexts. Those choices are indicated by reduced contrast.

menu bar. (1) The area near the top of a window, below the title bar and above the rest of the window, that contains choices that provide access to other menus. (2) In the AIX operating system, a rectangular area at the top of the client area of a window that contains the titles of the standard pull-down menus for that application.

message. (1) An assembly of characters and sometimes control codes that is transferred as an entity from an originator to one or more recipients. A message consists of two parts: envelope and content. (T) (2) A communication sent from a person or program to another person or program.

metropolitan area network (MAN). A network formed by the interconnection of two or more networks which may operate at higher speed than those networks, may cross administrative boundaries, and may use multiple access methods. (T) Contrast with *local area network (LAN)* and *wide area network (WAN)*.

MIB. (1) MIB module. (2) Management Information Base.

MIB object. Synonym for *MIB variable*.

MIB tree. In the Simple Network Management Protocol (SNMP), the structure of the Management Information Base (MIB).

MIB variable. In the Simple Network Management Protocol (SNMP), a specific instance of data defined in a MIB module. Synonymous with *MIB object*.

MIB view. In the Simple Network Management Protocol (SNMP), the collection of managed objects, known to the agent, that is visible to a particular community.

modem (modulator/demodulator). (1) A functional unit that modulates and demodulates signals. One of the functions of a modem is to enable digital data to be transmitted over analog transmission facilities. (T) (A) (2) A device that converts digital data from a computer to an analog signal that can be transmitted on a telecommunication line, and converts the analog signal received to data for the computer.

monitor. (1) A device that observes and records selected activities within a data processing system for analysis. Possible uses are to indicate significant departure from the norm, or to determine levels of utilization of particular functional units. (T) (2) Software or hardware that observes, supervises, controls, or verifies operations of a system. (A)

Motif. See *OSF/Motif*.

mouse. A commonly used pointing device, containing one or more buttons, with which a user can interact with a product or the operating environment.

mouse button. A mechanism on a mouse pointing device used to select objects or choices, initiate actions, or directly manipulate objects; a user presses a mouse button to interact with a computer system. The button makes a “clicking” sound when pressed and released.

Multiple Virtual Storage (MVS). See *MVS*.

multiport repeater. A repeater that contains multiple ports, for example, ThinLAN hubs or EtherTwist hubs.

MVS. Multiple Virtual Storage. Implies MVS/370, the MVS/XA product, and the MVS/ESA product.

N

navigate. In the NetView Graphic Monitor Facility, to move between levels in the view hierarchy.

navigation tree. In NetView for AIX, a component of the graphical user interface (GUI) that displays a hierarchy of open submaps illustrating the parent-child relationship. The navigation tree enables the network operator to determine which submaps are currently open and to close, restore, or raise the windows that contain submaps.

NETCENTER. A software product that assists the network operator and other technical personnel at a network control center in managing the network.

NetView for AIX. (1) Formerly known as *AIX SystemView NetView/6000* (or its abbreviated name, which is *AIX NetView/6000*). (2) An IBM licensed program for systems management in the AIX environment. NetView for AIX can use the NetView for AIX Service Point to communicate with the NetView and NETCENTER programs.

NetView for AIX Service Point. (1) Formerly known as the *AIX NetView Service Point*. (2) An IBM licensed program that operates in the AIX and UNIX environments. It functions as a gateway in an unattended environment.

NetView program. An IBM licensed program used to monitor and manage a network and to diagnose network problems.

network. (1) An arrangement of nodes and connecting branches. (T) (2) A configuration of data processing devices and software connected for information interchange. (3) A group of nodes and the links interconnecting them.

network address. (1) An identifier for a node, station, or unit of equipment in a network. (2) In a subarea network, an address, consisting of subarea and element fields, that identifies a link, link station, physical unit, logical unit, or system services control point. Subarea nodes use network addresses; peripheral nodes use local addresses or local-form session identifiers (LFSIDs). The boundary function in the subarea node to which a peripheral node is attached transforms local addresses or LFSIDs to network addresses and vice versa. Contrast with *network name*.

network administrator. A person who manages the use and maintenance of a network.

network class. In NetView for AIX, an object class used for symbols that represent compound objects that may contain objects such as hosts and network devices. Contrast with *connector class*.

Network File System (NFS). A protocol developed by Sun Microsystems, Incorporated, that allows any host in a network to mount another host's file directories. Once mounted, the file directory appears to reside on the local host.

network name. (1) The symbolic identifier by which end users refer to a network accessible unit, a link, or a link station within a given subnetwork. In APPN networks, network names are also used for routing purposes. Contrast with *network address*. (2) In a multiple-domain network, the name of the APPL statement defining a VTAM application program. The network name must be unique across domains. Contrast with *ACB name*. See *uninterpreted name*.

network node domain. An APPN network-node control point, its attached links, the network resources for which it answers directory search requests (namely, its local LUs and adjacent LEN end nodes), the adjacent APPN end nodes with which it exchanges directory search requests and replies, and other resources (such as a local storage device) associated with its own node or an adjacent end node for which it provides management services.

network operator. (1) A person who controls the operation of all or part of a network. (2) In a multiple-domain network, a person or program responsible for controlling all domains. (3) See also *domain operator*.

NFS. Network File System.

node. (1) In network topology, the point at an end of a branch. (T) (2) The representation of a state or an event by means of a point on a diagram. (A) (3) In a tree structure, a point at which subordinate items of data originate. (A) (4) An endpoint of a link or a junction common to two or more links in a network. Nodes can be processors, communication controllers, cluster controllers, or terminals. Nodes can vary in routing and other functional capabilities.

notification. (1) An unscheduled, spontaneously generated report of an event that has occurred. (2) In systems management, information emitted by a managed object relating to an event that has occurred within the managed object, such as a threshold violation or a change in configuration status.

O

object. (1) In object-oriented design or programming, an abstraction consisting of data and the operations associated with that data. See also *class*. (2) An item that a user can manipulate as a single unit to perform a task. An object can appear as text, an icon, or both.

object identifier. An administratively assigned data value of the type defined in abstract syntax notation 1 (ASN.1).

object registration service (ORS). In NetView for AIX, a component that creates and maintains a global directory of object managers, their locations, and their protocols. The postmaster daemon uses this directory to route messages and provide location transparency for managers and agents.

Off. A choice that appears in the cascaded menu from the Refresh choice. It sets the refresh function to off.

OK. A push button that accepts the information in a window and closes it. If the window contains changed information, those changes are applied before the window is closed.

On. A choice that appears in a cascaded menu from the Refresh choice. It immediately refreshes the view in a window.

online. (1) Pertaining to the operation of a functional unit when under the direct control of the computer. (T) (2) Pertaining to a user's ability to interact with a computer. (A)

Open. A choice that leads to a window in which users can select the object they want to open.

Open Software Foundation (OSF). A nonprofit research and development organization whose goals are (a) to develop specifications and software for use in an open software environment and (b) to make the specifications and software available to information technology vendors under fair and equitable licensing terms. For example, OSF developed the Distributed Computing Environment (DCE).

Open Systems Interconnection (OSI). The interconnection of open systems in accordance with standards of the International Organization for Standardization (ISO) for the exchange of information. (T) (A)

Open Systems Interconnection (OSI) architecture. Network architecture that adheres to that particular set of ISO standards that relates to Open Systems Interconnection. (T)

Open Systems Interconnection (OSI) reference model. A model that describes the general principles of the Open Systems Interconnection, as well as the purpose and the hierarchical arrangement of its seven layers. (T)

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)

operation. In object-oriented design or programming, a service that can be requested at the boundary of an object. Operations include modifying an object or disclosing information about an object.

operator. (1) A person or program responsible for managing activities controlled by a given piece of software such as MVS, the NetView program, or IMS. (2) A person who operates a device. (3) A person who keeps a system running.

ORS. Object registration service.

OS. Operating system.

OSF. Open Software Foundation.

OSF/Motif. A graphical interface that contains a toolkit, a presentation description language, a window manager, and a style guideline.

OSI. Open Systems Interconnection.

output. Pertaining to a device, process, or channel involved in an output process, or to the associated data or states. The word “output” may be used in place of “output data,” “output signal,” “output process,” when such a usage is clear in a given context. (T)

P

packet. In data communication, a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole. The data, control signals, and, possibly, error control information are arranged in a specific format. (I)

packet internet groper (PING). (1) In Internet communications, a program used in TCP/IP networks to test the ability to reach destinations by sending the destinations an Internet Control Message Protocol (ICMP) echo request and waiting for a reply. (2) In communications, a test of reachability.

page. (1) In a virtual storage system, a fixed-length block that has a virtual address and is transferred as a unit between real storage and auxiliary storage. (I) (A) (2) The information displayed at the same time on the screen of a display device.

panel. (1) See *window*. (2) A formatted display of information that appears on a display screen. See *help panel* and *task panel*. (3) In computer graphics, a display image that defines the locations and characteristics of display fields on a display surface.

Paste. A choice that places the contents of the clipboard at the current cursor position.

path. The route used to locate files; the storage location of a file. A fully qualified path lists the drive identifier, directory name, subdirectory name (if any), and file name with the associated extension.

physical connection. (1) A connection that establishes an electrical circuit. (2) A point-to-point or multipoint connection. (3) Synonymous with *connection*.

PING. Packet internet groper.

point-to-point. Pertaining to data transmission between two locations without the use of any intermediate display station or computer.

pointer. (1) A data element that indicates the location of another data element. (T) (2) An identifier that indicates the location of an item of data. (A)

polling. (1) On a multipoint connection or a point-to-point connection, the process whereby data stations are invited, one at a time, to transmit. (I) (2) Interrogation of devices for such purposes as to avoid contention, to determine operational status, or to determine readiness to send or receive data. (A)

pop-up menu. A menu that, when requested, appears next to the object it is associated with.

port. (1) An access point for data entry or exit. (2) A connector on a device to which cables for other devices such as display stations and printers are attached. (3) The representation of a physical connection to the link hardware. A port is sometimes referred to as an adapter; however, there can be more than one port on an adapter. There may be one or more ports controlled by a single DLC process. (4) In the Internet suite of protocols, a 16-bit number used to communicate between TCP or the User Datagram Protocol (UDP) and a higher-level protocol or application. Some protocols, such as File Transfer Protocol (FTP) and Simple Mail Transfer Protocol (SMTP), use the same well-known port number in all TCP/IP implementations. (5) An abstraction used by transport protocols to distinguish among multiple destinations within a host machine. (6) Synonymous with *socket*.

primary window. In OSF/Motif, the top-level window in an application program that can be minimized or represented by an icon. See also *submap window*.

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, environmental failure such as a power loss, or user error.

processor. In a computer, a functional unit that interprets and executes instructions. A processor consists of at least an instruction control unit and an arithmetic and logic unit. (T)

protocol. (1) A set of semantic and syntactic rules that determine the behavior of functional units in achieving communication. (I) (2) In Open Systems Interconnection architecture, a set of semantic and syntactic rules that determine the behavior of entities in the same layer in performing communication functions. (T)

proxy agent. A process or entity that is both an agent to its manager and a manager for one or more objects. It satisfies requests from its manager by relaying those requests and translating them for the objects that it manages.

pull-down menu. See *menu*.

push button. A button, labeled with text, graphics, or both, that represents an action that will be initiated when a user selects it.

Q

queue. (1) A list constructed and maintained so that the next data element to be retrieved is the one stored first. (T) (2) A line or list of items waiting to be processed; for example, work to be performed or messages to be displayed. (3) To arrange in or form a queue.

R

radio button. A circle with text beside it. Radio buttons are combined to show a user a fixed set of choices from which the user can select one. The circle becomes partially filled when a choice is selected.

RARP. Reverse Address Resolution Protocol.

read-only. A type of access to data that allows data to be read but not copied, printed, or modified.

real time. (1) In Open Systems Interconnection architecture, pertaining to the processing of data by a computer in connection with another process outside the computer according to time requirements imposed by the outside process. This term is also used to describe systems operating in conversational mode and processes that can be influenced by human intervention while they are in progress. (I) (A) (2) In Open Systems Interconnection architecture, pertaining to an application such as a process control system or a computer-assisted instruction system in which response to input is fast enough to affect subsequent input.

recommended action. Procedures suggested by the NetView program that can be used to determine the causes of network problems.

record. (1) In programming languages, an aggregate that consists of data objects, possibly with different attributes, that usually have identifiers attached to them. In some programming languages, records are called structures. (I) (2) A set of data treated as a unit. (T) (3) A set of one or more related data items grouped for processing.

recording filter. In the NetView program, the function that determines which events, statistics, and alerts are stored on a database.

reduced instruction-set computer (RISC). A computer that uses a small, simplified set of frequently used instructions for rapid execution.

Refresh. A cascading choice that gives a user access to other choices (*On* and *Off*) that control whether changes made to underlying data in a window are displayed immediately, not displayed at all, or displayed at a later time.

registration file. See *application registration file*, *field registration file*, *local registration file (LRF)*, and *symbol registration file*.

relation. In a relational database, a set of entity occurrences that have the same attributes. (T)

relational database. A database in which the data are organized and accessed according to relations. (T)

remote. (1) Pertaining to a system, program, or device that is accessed through a telecommunication line. (2) Contrast with *local*.

repeater. A node of a local area network, a device that regenerates signals in order to extend the range of transmission between data stations or to interconnect two branches. (T)

Request for Comments (RFC). In Internet communications, the document series that describes a part of the Internet suite of protocols and related experiments. All Internet standards are documented as RFCs.

resource. Any facility of a 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.

response. (1) In data communication, a reply represented in the control field of a response frame. It advises the primary or combined station of the action taken by the secondary or other combined station to one or more commands. (2) See also *command*.

response time. For response time monitoring, the time from the activation of a transaction until a response is received, according to the response time definition coded in the performance class.

Reverse Address Resolution Protocol (RARP). (1) In the Internet suite of protocols, the protocol that maps a hardware (MAC) address to an IP address. RARP can be used to determine a port's IP address. (2) See also *Address Resolution Protocol (ARP)*.

RFC. Request for Comments.

ring. See *ring network*.

ring network. (1) A network in which every node has exactly two branches connected to it and in which there are exactly two paths between any two nodes. (T) (2) A network configuration in which devices are connected by unidirectional transmission links to form a closed path.

RISC. Reduced instruction-set computer.

root user. See *superuser authority.*

route. (1) An ordered sequence of nodes and transmission groups (TGs) that represent a path from an origin node to a destination node traversed by the traffic exchanged between them. (2) The path that network traffic uses to get from source to destination.

router. (1) A computer that determines the path of network traffic flow. The path selection is made from several paths based on information obtained from specific protocols, algorithms that attempt to identify the shortest or best path, and other criteria such as metrics or protocol-specific destination addresses. (2) An attaching device that connects two LAN segments, which use similar or different architectures, at the reference model network layer. (3) In OSI terminology, a function that determines a path by which an entity can be reached. (4) In TCP/IP, synonymous with *gateway*. (5) Contrast with *bridge*.

routing. (1) The process of determining the path to be used for transmission of a message over a network. (T) (2) The assignment of the path by which a message is to reach its destination.

S

screen. (1) The physical surface of a display device upon which information is shown to users. (2) In the AIX extended curses library, a window that is as large as the display screen of the workstation. (3) Deprecated term for *display panel*.

scroll. To move a display image vertically or horizontally to view data that otherwise cannot be observed within the boundaries of the display screen.

scroll bar. A window component that shows a user that more information is available in a particular direction and can be scrolled into view. Scroll bars can be either horizontal or vertical.

seed file. In NetView for AIX, a file that contains a list of nodes within an Administrative Domain, which the automatic discovery function uses to accelerate the generation of the network topology map.

segment. (1) A portion of a computer program that may be executed without the entire computer program being resident in main storage. (T) (2) A group of display elements. (3) A section of cable between components or devices. A segment may consist of a single patch cable, several patch cables that are connected, or a combination of building cable and patch cables that are connected. (4) In the Enhanced X-Windows Toolkit, one or more lines that are drawn but not necessarily connected at the endpoints. (5) In LANs or WANs, a subset of nodes in a network or subnet that are connected by a common physical medium.

select. To explicitly identify one or more objects to which a subsequent choice will apply.

selection. The process of explicitly identifying one or more objects to which a subsequent choice will apply.

server. (1) A functional unit that provides shared services to workstations over a network; for example, a file server, a print server, a mail server. (T) (2) In a network, a data station that provides facilities to other stations; for example, a file server, a print server, a mail server. (A) (3) In the AIX operating system, an application program that usually runs in the background and is controlled by the system program controller. (4) In the Enhanced X-Windows Toolkit, a program that provides the basic windowing mechanism. It handles interprocess communication (IPC) connections from clients, demultiplexes graphics requests onto screens, and multiplexes input back to clients.

service point (SP). (1) An entry point that supports applications that provide network management for resources not under the direct control of itself as an entry point. Each resource is either under the direct control of another entry point or not under the direct control of any entry point. A service point accessing these resources is not required to use SNA sessions (unlike a focal point). A service point is needed when entry point support is not yet available for some network management function. (2) In NetView for AIX, see *NetView for AIX Service Point*.

shared. Pertaining to the availability of a resource for more than one use at the same time.

shell procedure. In the AIX operating system, a series of commands, combined in a file, that carry out a particular function when the file is run or when the file is spec-

ified as a value to the SH command. Synonymous with *shell script*.

shell script. Synonym for *shell procedure*.

shutdown. The process of ending operation of a system or a subsystem, following a defined procedure.

Simple Network Management Protocol (SNMP). In the Internet suite of protocols, a network management protocol that is used to monitor routers and attached networks. SNMP is an application layer protocol. Information on devices managed is defined and stored in the application's Management Information Base (MIB).

SMUX. SNMP multiplexer.

SNA. Systems Network Architecture.

snapshot. In NetView for AIX, a copy of a map that reflects the topology and status of the map's nodes and links at a given moment in time.

SNMP. Simple Network Management Protocol.

SNMP multiplexer (SMUX). A protocol that is used by a subagent to provide local and remote system monitoring using the Simple Network Management Protocol (SNMP).

socket. (1) An endpoint for communication between processes or application programs. (2) Synonym for *port*.

softcopy. (1) A nonpermanent copy of the contents of storage in the form of a display image. (T) (2) One or more files that can be electronically distributed, manipulated, and printed by a user. (3) Contrast with *hardcopy*.

spin button. A component used to display, in sequence, a ring of related but mutually exclusive choices. A user can accept the value displayed in the entry field or can type a valid choice into the entry field.

SSCP. System services control point.

static. (1) In programming languages, pertaining to properties that can be established before execution of a program; for example, the length of a fixed length variable is static. (I) (2) Pertaining to an operation that occurs at a predetermined or fixed time. (3) Contrast with *dynamic*.

station. An input or output point of a system that uses telecommunication facilities; for example, one or more systems, computers, terminals, devices, and associated programs at a particular location that can send or receive data over a telecommunication line.

status. The condition or state of hardware or software, usually represented by a status code.

subagent. In the Simple Network Management Protocol (SNMP), something that provides an extension to the utility provided by the SNMP agent.

submap. In NetView for AIX, a particular view of some aspect of a network that displays symbols representing objects. The application program that creates a submap determines what part of the network the submap displays.

submap stack. In NetView for AIX, a component of the graphical user interface shown on the left side of each submap window. The submap stack represents the navigational path used to reach the particular submap, and it can be used to select a previously viewed submap.

submap window. In NetView for AIX, the graphical component that contains a menu bar, a submap viewing area, a status line, and a button box. A user can display multiple submap windows of an open map and an open snapshot at any given time. See also *primary window*.

subnet. (1) In TCP/IP, a part of a network that is identified by a portion of the IP address. (2) Synonym for *subnetwork*.

subnet mask. Synonym for *address mask*.

subnetwork. (1) Any group of nodes that have a set of common characteristics, such as the same network ID. (2) In the AIX operating system, one of a group of multiple logical network divisions of another network, such as can be created by the Transmission Control Protocol/Internet Protocol (TCP/IP) interface program. (3) Synonymous with *subnet*.

subnetwork mask. Synonym for *address mask*.

subvector. A subcomponent of the network management vector transport (NMVT) major vector.

superuser authority. In the AIX operating system, the unrestricted authority to access and modify any part of the operating system, usually associated with the user who manages the system.

symbol. In NetView for AIX, a picture or an icon on a submap that represents an object (a network resource or an application). Each symbol belongs to a class, repres-

ented by the symbol's shape, and to a subclass, represented by the design within the shape. The symbol reflects characteristics of the object it represents, such as its status; it also has characteristics of its own, such as behavior.

symbol registration file. In NetView for AIX, a file used to define symbol classes and subclasses.

system services control point (SSCP). A component within a subarea network for managing the configuration, coordinating network operator and problem determination requests, and providing directory services and other session services for end users of the network. Multiple SSCPs, cooperating as peers with one another, 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 own domain.

system services control point (SSCP) domain. The system services control point, the physical units (PUs), the logical units (LUs), the links, the link stations, and all the resources that the SSCP has the ability to control by means of activation and deactivation requests.

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. The layered structure of SNA allows the ultimate origins and destinations of information, that is, the end users, to be independent of and unaffected by the specific SNA network services and facilities used for information exchange.

T

task. In a multiprogramming or multiprocessing environment, one or more sequences of instructions treated by a control program as an element of work to be accomplished by a computer. (I) (A)

task panel. Online display from which you communicate with the program in order to accomplish the program's function, either by selecting an option provided on the panel or by entering an explicit command. See also *help panel*.

TCP. Transmission Control Protocol.

TCP/IP. Transmission Control Protocol/Internet Protocol.

Telnet. In the Internet suite of protocols, a protocol that provides remote terminal connection service. It allows users of one host to log on to a remote host and interact as directly attached terminal users of that host.

terminal. A device, usually equipped with a keyboard and a display device, that is capable of sending and receiving information.

threshold. In NetView for AIX, a setting that specifies the maximum value a statistic can reach before notification that the limit was exceeded. For example, when a monitored MIB value has exceeded the threshold, the data collector generates a threshold event.

timeout. (1) An event that occurs at the end of a predetermined period of time that began at the occurrence of another specified event. (I) (2) A time interval allotted for certain operations to occur; for example, response to polling or addressing before system operation is interrupted and must be restarted.

toggle. (1) Pertaining to any device having two stable states. (A) (2) Pertaining to a switching device, such as a toggle key on a keyboard, that allows a user to switch between two types of operations.

toggle button. In the AIXwindows Toolkit and the Enhanced X-Windows Toolkit, a graphical object that simulates a toggle switch; it switches sequentially from one optional state to another.

token. (1) In a local area network, the symbol of authority passed successively from one data station to another to indicate the station temporarily in control of the transmission medium. Each data station has an opportunity to acquire and use the token to control the medium. A token is a particular message or bit pattern that signifies permission to transmit. (T) (2) In LANs, a sequence of bits passed from one device to another along the transmission medium. When the token has data appended to it, it becomes a frame.

token ring. (1) According to IEEE 802.5, network technology that controls media access by passing a token (special packet or frame) between media-attached stations. (2) A FDDI or IEEE 802.5 network with a ring topology that passes tokens from one attaching ring station (node) to another. (3) See also *local area network (LAN)*.

tool palette. In NetView for AIX, a component of the graphical user interface (GUI) that enables the network operator to open application program instances by using

the mouse to drag-and-drop the icons that represent the application program.

topology. In communications, the physical or logical arrangement of nodes in a network, especially the relationships among nodes and the links between them.

trace. A record of the execution of a computer program. It exhibits the sequences in which the instructions were executed. (A)

transaction program (TP). (1) A program that processes transactions in an SNA network. There are two kinds of transaction programs: application transaction programs and service transaction programs. See also *conversation*. (2) In VTAM, 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 application program, using protocols defined by that application program. The application program, in turn, could request services from VTAM by issuing the APPCCMD macroinstruction.

Transmission Control Protocol (TCP). A communications protocol used in the Internet and in any network that follows the U.S. Department of Defense standards for internetwork protocol. TCP provides a reliable host-to-host protocol between hosts in packet-switched communications networks and in interconnected systems of such networks. It uses the Internet Protocol (IP) as the underlying protocol.

Transmission Control Protocol/Internet Protocol (TCP/IP). A set of communications protocols that support peer-to-peer connectivity functions for both local and wide area networks.

trap. In the Simple Network Management Protocol (SNMP), a message sent by a managed node (agent function) to a management station to report an exception condition.

tree structure. A data structure that represents entities in nodes, with at most one parent node for each node, and with only one root node. (T)

U

UDP. User Datagram Protocol.

uninterpreted name. In SNA, a character string that a system services control point (SSCP) can convert into the network name of a logical unit (LU). Typically, an uninterpreted name is used in a logon or Initiate request from a secondary logical unit (SLU) to identify the primary logical unit (PLU) with which the session is requested.

user. (1) Any person or any thing that may issue or receive commands and messages to or from the information processing system. (T) (2) Anyone who requires the services of a computing system.

User Datagram Protocol (UDP). In the Internet suite of protocols, a protocol that provides unreliable, connectionless datagram service. It enables an application program on one machine or process to send a datagram to an application program on another machine or process. UDP uses the Internet Protocol (IP) to deliver datagrams.

user plane. In NetView for AIX, the submap layer on which symbols of objects that are not managed by an application program are displayed. Symbols on the user plane are displayed with a shadow, which makes them appear higher than symbols on the application plane. See also *background plane*.

V

value. (1) A specific occurrence of an attribute; for example, “blue” for the attribute “color.” (T) (2) A quantity assigned to a constant, a variable, a parameter, or a symbol.

variable. (1) In programming languages, a language object that may take different values, one at a time. The values of a variable are usually restricted to a certain data type. (I) (2) A quantity that can assume any of a given set of values. (A) (3) A name used to represent a data item whose value can be changed while the program is running. (4) In the Simple Network Management Protocol (SNMP), a match of an object instance name with an associated value.

version. A separately licensed program that usually has significant new code or new function.

vertex. In graphs, a point that may be the end of an arc or the intersection of multiple arcs.

viewing filter. In the NetView program, the function that allows a user to select the alert data to be displayed on a terminal. All other stored data is blocked.

virtual machine (VM). (1) A virtual data processing system that appears to be at the exclusive disposal of a particular user, but whose functions are accomplished by sharing the resources of a real data processing system. (T) (2) In VM/ESA, the virtual processors, virtual storage, virtual devices, and virtual channel subsystem allocated to a single user. A virtual machine also includes any expanded storage dedicated to it.

VM. Virtual machine.

W

WAN. Wide area network.

wide area network (WAN). (1) A network that provides communication services to a geographic area larger than that served by a local area network or a metropolitan area network, and that may use or provide public communication facilities. (T) (2) A data communication network designed to serve an area of hundreds or thousands of miles; for example, public and private packet-switching networks, and national telephone networks. (3) Contrast with *local area network (LAN)* and *metropolitan area network (MAN)*.

window. (1) A portion of a display surface in which display images pertaining to a particular application can be presented. Different applications can be displayed simultaneously in different windows. (A) (2) An area with visible boundaries that presents a view of an object or with which a user conducts a dialog with a computer system.

workstation. (1) A functional unit at which a user works. A workstation often has some processing capability. (T) (2) One or more programmable or nonprogrammable devices that allow a user to do work. (3) A terminal or microcomputer, usually one that is connected to a mainframe or to a network, at which a user can perform applications.

write access. In computer security, permission to write to an object.

X

X Window System. A software system, developed by the Massachusetts Institute of Technology, that allows the user of a display to concurrently use multiple application programs through different windows of the display. The application programs may execute on different computers.

Z

zoom. In CUA architecture, to progressively increase or decrease the size of a part of an image on a screen or in a window.

Bibliography

NetView for AIX Publications

The following paragraphs briefly describe the publications for Version 4 of the NetView for AIX program:

NetView for AIX Concepts: A General Information Manual (GC31-8160)

This book provides an overview of the NetView for AIX program that business executives can use to evaluate the product. System planners can also use this information to learn how NetView for AIX manages heterogeneous networks.

NetView for AIX Database Guide (SC31-8167)

This book provides information for system administrators and database administrators to configure the NetView for AIX program to work with the following relational database management systems: DB2/6000, INFORMIX, INGRES, ORACLE, and SYBASE. This book also describes how to transfer IP topology, trapdlog, and snmpCollect data to the relational database and how to manipulate the data.

NetView for AIX Installation and Configuration (SC31-8163)

This book provides installation and configuration steps for the system programmer who will install and configure the NetView for AIX program.

NetView for AIX User's Guide for Beginners (SC31-8158)

This book contains “how-to” information that provides network operators the help they need to get acquainted with NetView for AIX and accomplish some basic networking tasks. It is written for the user who is unfamiliar with the NetView for AIX program.

NetView for AIX Administrator's Guide (SC31-8168)

This book explains network management principles and describes how the NetView for AIX program's components work together. It is for the advanced user. Most of the tasks require root authority. This book includes tasks such as customizing the graphical interface, filtering events, configuring events, and managing network performance and configuration.

NetView for AIX Administrator's Reference (SC31-8169)

This book contains reference information for commands, daemons, and files. It is used primarily when performing administrative tasks.

NetView for AIX Diagnosis Guide (SC31-8162)

This book is intended to help you classify and resolve problems related to the operation of the NetView for AIX program.

NetView for AIX Application Interface Style Guide (SC31-6240)

This book provides guidelines for system programmers who develop applications that will be integrated with the NetView for AIX program.

NetView for AIX Programmer's Guide (SC31-8164)

This book provides information for programmers about creating network management applications. This book also contains information about the NetView for AIX program server, commands, function calls, and object classes.

NetView for AIX Programmer's Reference (SC31-8165)

This book is intended for programmers and contains reference information about the NetView for AIX program and its server, commands, function calls, and object classes.

NetView for AIX and the Host Connection (SC31-8161)

This book provides information for System/390 and NetView users who want to manage TCP/IP and SNA networks.

Quick Reference Card (SX75-0113)

This summary provides a brief description of each NetView for AIX daemon. The card also lists the menu items and the submenu items below them.

In addition to these printed books, online documentation of the NetView for AIX library is available. An online Help Index is also available from the NetView for AIX Help pull-down window. The Help Index provides dialog box help and task help.

IBM RISC System/6000 Publications

In addition to the NetView for AIX documentation, the following publications may also be helpful to users:

AIX Quick Reference (SC23-2401)

Task Index and Glossary for IBM RISC System/6000 (GC23-2201)

IBM RISC System/6000 Problem Solving Guide (SC23-2204)

AIX Communications Concepts and Procedures for IBM RISC System/6000 (GC23-2203)

AIX Commands Reference for IBM RISC System/6000 (GC23-2366, GC23-2367, GC23-2376, GC23-2393)

AIX Files Reference for IBM RISC System/6000 (GC23-2200)

NetView Publications

The following list contains selected NetView Version 2 Release 3 publications:

NetView Administration Reference (SC31-6128)

NetView At a Glance (GC31-7016)

NetView Automation Planning (SC31-6141)

NetView Customization Guide (SC31-6132)

NetView Installation and Administration Guide (MVS: SC31-6125) (VM: SC31-6182) (VSE: SC31-6182)

NetView Operation (SC31-6127)

NetView Problem Determination and Diagnosis (LY43-0014)

NetView Resource Alerts Reference (SC31-6136)

NetView Samples (MVS: SC31-6126) (VM: SC31-6183) (VSE: SC31-6184)

The following list contains selected NetView Version 2 Release 4 publications:

NetView Administration Reference (SC31-7080)

NetView Automation Planning (SC31-7082)

NetView Customization Guide (SC31-7091)

NetView General Information (GC31-7098)

NetView Installation and Administration Facility/2 Guide (SC31-7099)

NetView Installation and Administration Guide (SC31-7084)

NetView Operation (SC31-7066)

NetView Problem Determination and Diagnosis (LY43-0101)

NetView Resource Alerts Reference (SC31-7097)

TCP/IP Publications for AIX (RS/6000, PS/2, RT, 370)

The following list shows the books available for TCP/IP in the AIX Operating System library:

AIX Operating System TCP/IP User's Guide (SC23-2309)

AIX PS/2 TCP/IP User's Guide (SC23-2047)

TCP/IP for IBM X-Windows on DOS (SC23-2349)

AIX SNA Services/6000 Publications

The following list of publications are for use with the AIX Operating System:

AIX SNA Server/6000 User's Guide (SC31-7002)

AIX SNA Server/6000 Configuration Reference (SC31-7014)

AIX SNA Server/6000 Transaction Program (SC31-7003)

Internet Request for Comments (RFCs)

The following documents describe Internet standards supported by the NetView for AIX program. Copies of these documents are shipped on the AIX SystemView NetView/6000 product installation media. They are installed in the /usr/OV/doc directory.

RFC 1095: The Common Management Services and Protocol over TCP/IP (CMOT)

RFC 1155: Structure and Identification of Management Information for TCP/IP-Based Internets

RFC 1157: Simple Network Management Protocol (SNMP)

RFC 1187: Bulk Table Retrieval with the SNMP

RFC 1189: The Common Management Information Services and Protocols for the Internet (CMOT and CMIP)

RFC 1212: Concise MIB Definitions

RFC 1213: Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II

RFC 1215: Convention for Defining Traps for Use with the SNMP

RFC 1229: Extensions to the Generic-Interface MIB

RFC 1230: IEEE 802.4 Token Bus MIB

RFC 1231: IEEE 802.5 Token Bus MIB

RFC 1232: Definitions of Managed Objects for the DS1 Interface Type

RFC 1233: Definitions of Managed Objects for the DS3 Interface Type

RFC 1239: Reassignment of Experimental MIBs to Standard MIBs

RFC 1243: AppleTalk Management Information Base

RFC 1253: OSPF Version 2 Management Information Base

RFC 1269: Definitions of Managed Objects for the Border Gateway Protocol (Version 3)

RFC 1271: Remote Network Monitoring Management Information Base

RFC 1284: Definitions of Managed Objects for the Ethernet-like Interface Types

RFC 1285: FDDI Management Information Base

RFC 1286: Definitions of Managed Objects for Bridges

RFC 1289: DECnet Phase IV MIB Extensions

RFC 1304: Definition of Managed Objects for the SIP Interface Type

RFC 1315: Management Information Base for Frame Relay DTEs

RFC 1316: Definitions of Managed Objects for Character Stream Devices

RFC 1317: Definitions of Managed Objects for RS-232-like Hardware Devices

RFC 1318: Definitions of Managed Objects for Parallel-printer-like Hardware Devices

RFC 1450: Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)

RFC 1452: Coexistence between Version 1 and Version 2 of the Internet-Standard Network Management Framework

Related Publications

The following publications are closely related to or referenced by the NetView for AIX Library:

AIX Trouble Ticket/6000 Publications

For information about the AIX Trouble Ticket/6000 program, consult the following publications:

AIX Trouble Ticket/6000 Brochure (GC31-7161)

AIX Trouble Ticket/6000 User's Guide (SC31-7162)

Service Point Publication

AIX NetView Service Point Installation, Operation, and Programming Guide (SC31-6120)

Other IBM TCP/IP Publications

The following list shows other available IBM TCP/IP publications:

Introducing IBM Transmission Control Protocol/Internet Protocol Products for OS/2, VM, and MVS (GC31-6080)

IBM TCP/IP Version 2 for VM and MVS: Diagnosis Guide (LY43-0013)

MVS/DFP Version 3 Release 3: Using the Network File System Server (SC26-4732)

SNMP Information

You can use the following sources for detailed SNMP information:

The Simple Book, M.T. Rose, Prentice-Hall, 1991 (ISBN 0-13-812611-9)

The *Windows SNMP Manager API Specification*, the *WinSNMP/MIB API Specification*, and other information on Windows SNMP are available through anonymous FTP from the host sunsite.unc.edu under the directory path /pub/micro/pc-stuff/ms-windows/WinSNMP

These Internet standards provide SNMP information:

RFC 1901: Introduction to Community-based SNMPv2

RFC 1902: Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)

RFC 1903: Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)

RFC 1904: Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2)

RFC 1905: Protocol Operation for Version 2 of the Simple Network Management Protocol (SNMPv2)

RFC 1906: Transport Mapping for Version 2 of the Simple Network Management Protocol (SNMPv2)

RFC 1907: Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)

RFC 1908: Coexistence between Version 1 and Version 2 of Internet-standard Network Management Framework

RFC 1909: An Administrative Infrastructure for SNMPv2 (SNMPv2USEC)

RFC 1910: User-based Security Model for SNMPv2 (SNMPv2USEC)

X Window System Publications

The following list shows selected X Window System publications:

Introduction to the X Window System, Oliver Jones, Prentice-Hall, 1988 (ISBN 0-13-499997)

X Window System Technical Reference, Steven Mikes, Addison-Wesley, 1990 (ISBN 0-201-52370)

X Window System: Programming and Applications with Xt, Douglas A. Young, Prentice-Hall, 1989 (ISBN 0-13-972167)

X Window System: Programming and Applications with Xt, OSF/Motif Edition, Douglas A. Young, Prentice-Hall, 1990 (ISBN 0-13-497074)

X/Open Specification

For information about the X/Open OSI-Abstract-Data Manipulation (XOM) application programming interface (API), consult the following X/Open documents:

X/Open OSI-Abstract-Data Manipulation (XOM) API, CAE Specification

X/Open Preliminary Specification. Systems Management: GDMO to XOM Translation Algorithm

OSF/Motif Publications

The following list contains selected OSF/Motif publications:

OSF/Motif Series (5 volumes), Open Software Foundation, Prentice Hall, Inc. 1990

OSF/Motif Application Environment Specifications, (AES) (ISBN 0-13-640483-9)

OSF/Motif Programmer's Guide (ISBN 0-13-640509-6)

OSF/Motif Programmer's Reference, (ISBN 0-13-640517-7)

OSF/Motif Style Guide (ISBN 0-13-640491-X)

OSF/Motif User's Guide, (ISBN 0-13-640525-8)

ISO/IEC Standards

For information about the ISO/IEC standards on which the NetView for AIX program is based, refer to the following publications:

ISO IS 7498-4, Open Systems Interconnection–Basic Reference Model–Part 4: Management Framework

ISO 8824, Open Systems Interconnection–Specification of Abstract Syntax Notation One (ASN.1)

ISO 8825, Open Systems Interconnection– Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)

ISO IS 9595, Common Management Information–Service Definition

ISO IS 9596-1, Common Management Information–Protocol Specification

ISO DIS 9899, Information Processing–Programming Language C

ISO 10040, Systems Management Overview

The ISO/IEC standards can be obtained from the following address:

OMNICOM
243 Church St. NW
Vienna, VA 22180-4434

(800) OMNICOM
(703) 281-1135
(703) 281-1505 (FAX)

Index

Special Characters

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

Communicating Your Comments to IBM

NetView for AIX
Administrator's Guide
Version 4

Publication No. SC31-8168-01

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM 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.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:

  United States and Canada: **1-800-227-5088**
- If you prefer to send comments electronically, use this network ID:
  - IBM Mail Exchange: **USIB2HPD at IBMMAIL**
  - IBMLink: **CIBMORCF at RALVM13**
  - Internet: **USIB2HPD@VNET.IBM.COM**

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Help us help you!

NetView for AIX
Administrator's Guide
Version 4

Publication No. SC31-8168-01

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	□	□

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	□	□
Complete	□	□
Easy to find	□	□
Easy to understand	□	□
Well organized	□	□
Applicable to your task	□	□

Specific Comments or Problems:

Please tell us how we can improve this book:

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

Your Internet Address: _____

Name _____ Address _____

Company or Organization

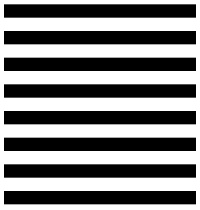
Phone No.

Help us help you!
SC31-8168-01

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

Information Development
Department CGMD
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

Fold and Tape **Please do not staple** Fold and Tape

SC31-8168-01

Cut or Fold
Along Line

Program Number: 5765-527

Printed in U.S.A.

SC31-8168-01

DSMKPO653E POSTSCRIPT FILE '@E@P@S' NOT FOUND.
DSMMOM395I '.EDFPO' LINE 70: .po @E@P@S
DSMMOM397I '.EDFPO' WAS IMBEDDED AT LINE 910 OF '.EDFAWRK'
DSMMOM397I '.EDFAWRK' WAS IMBEDDED AT LINE 2330 OF '.EDF#CV'
DSMMOM397I '.EDF#CV' WAS IMBEDDED AT LINE 120 OF '.EDF#FCV5'
DSMMOM397I '.EDF#FCV5' WAS IMBEDDED AT LINE 330 OF '.EDFCOVER'
DSMMOM397I '.EDFCOVER' WAS IMBEDDED AT LINE 50 OF 'LBZL1MST'
+++EDF030W ARTWORK REFID=LBZL0C2 does not match any ID on ARTDEF or OVERLAY tag. (Page 290 File: LBZL1C2)
DSMMOM397I '.EDF@RFID' WAS IMBEDDED AT LINE 50 OF '.EDFAWRK'
DSMMOM397I '.EDFAWRK' WAS IMBEDDED AT LINE 2 OF 'LBZL1C2'
DSMMOM397I 'LBZL1C2' WAS IMBEDDED AT LINE 187 OF 'EDFPRF40'
DSMBEG323I STARTING PASS 2 OF 4.
DSMKPO653E POSTSCRIPT FILE '@E@P@S' NOT FOUND.
DSMMOM395I '.EDFPO' LINE 70: .po @E@P@S
DSMMOM397I '.EDFPO' WAS IMBEDDED AT LINE 910 OF '.EDFAWRK'
DSMMOM397I '.EDFAWRK' WAS IMBEDDED AT LINE 2330 OF '.EDF#CV'
DSMMOM397I '.EDF#CV' WAS IMBEDDED AT LINE 120 OF '.EDF#FCV5'
DSMMOM397I '.EDF#FCV5' WAS IMBEDDED AT LINE 330 OF '.EDFCOVER'
DSMMOM397I '.EDFCOVER' WAS IMBEDDED AT LINE 50 OF 'LBZL1MST'
+++EDF030W ARTWORK REFID=LBZL0C2 does not match any ID on ARTDEF or OVERLAY tag. (Page 290 File: LBZL1C2)
DSMMOM397I '.EDF@RFID' WAS IMBEDDED AT LINE 50 OF '.EDFAWRK'
DSMMOM397I '.EDFAWRK' WAS IMBEDDED AT LINE 2 OF 'LBZL1C2'
DSMMOM397I 'LBZL1C2' WAS IMBEDDED AT LINE 187 OF 'EDFPRF40'
DSMBEG323I STARTING PASS 3 OF 4.
DSMKPO653E POSTSCRIPT FILE '@E@P@S' NOT FOUND.
DSMMOM395I '.EDFPO' LINE 70: .po @E@P@S
DSMMOM397I '.EDFPO' WAS IMBEDDED AT LINE 910 OF '.EDFAWRK'
DSMMOM397I '.EDFAWRK' WAS IMBEDDED AT LINE 2330 OF '.EDF#CV'
DSMMOM397I '.EDF#CV' WAS IMBEDDED AT LINE 120 OF '.EDF#FCV5'
DSMMOM397I '.EDF#FCV5' WAS IMBEDDED AT LINE 330 OF '.EDFCOVER'
DSMMOM397I '.EDFCOVER' WAS IMBEDDED AT LINE 50 OF 'LBZL1MST'
+++EDF030W ARTWORK REFID=LBZL0C2 does not match any ID on ARTDEF or OVERLAY tag. (Page 290 File: LBZL1C2)
DSMMOM397I '.EDF@RFID' WAS IMBEDDED AT LINE 50 OF '.EDFAWRK'
DSMMOM397I '.EDFAWRK' WAS IMBEDDED AT LINE 2 OF 'LBZL1C2'
DSMMOM397I 'LBZL1C2' WAS IMBEDDED AT LINE 187 OF 'EDFPRF40'
DSMBEG323I STARTING PASS 4 OF 4.
DSMKPO653E POSTSCRIPT FILE '@E@P@S' NOT FOUND.
DSMMOM395I '.EDFPO' LINE 70: .po @E@P@S
DSMMOM397I '.EDFPO' WAS IMBEDDED AT LINE 910 OF '.EDFAWRK'
DSMMOM397I '.EDFAWRK' WAS IMBEDDED AT LINE 2330 OF '.EDF#CV'
DSMMOM397I '.EDF#CV' WAS IMBEDDED AT LINE 120 OF '.EDF#FCV5'
DSMMOM397I '.EDF#FCV5' WAS IMBEDDED AT LINE 330 OF '.EDFCOVER'
DSMMOM397I '.EDFCOVER' WAS IMBEDDED AT LINE 50 OF 'LBZL1MST'
+++EDF099W RCFTEXT information too long for form. (Page 305 File: LBZL1MST SCRIPT)
DSMMOM397I '.EDF#RCFF' WAS IMBEDDED AT LINE 2250 OF '.EDF#RCF'
DSMMOM397I '.EDF#RCF' WAS IMBEDDED AT LINE 140 OF '.EDFEDTYP'
DSMMOM397I '.EDFEDTYP' WAS IMBEDDED AT LINE 109 OF 'LBZL1MST'
+++EDF030W ARTWORK REFID=LBZL0C2 does not match any ID on ARTDEF or OVERLAY tag. (Page 308 File: LBZL1C2)
DSMMOM397I '.EDF@RFID' WAS IMBEDDED AT LINE 50 OF '.EDFAWRK'
DSMMOM397I '.EDFAWRK' WAS IMBEDDED AT LINE 2 OF 'LBZL1C2'
DSMMOM397I 'LBZL1C2' WAS IMBEDDED AT LINE 187 OF 'EDFPRF40'