HP OpenView
Service Information Portal 2.0

Presenting NNM Data

Windows NT®, Windows® 2000, HP-UX, and Solaris

Manufacturing Part Number: J4797-90003
April 2001

Legal Notices

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty. A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

HEWLETT-PACKARD COMPANY

3404 E. Harmony Road

Fort Collins, CO 80528 U.S.A.

Use of this manual and flexible disk(s), tape cartridge(s), or CD-ROM(s) supplied for this pack is restricted to this product only. Additional copies of the programs may be made for security and back-up purposes only. Resale of the programs in their present form or with alterations, is expressly prohibited.

Trademark Notices.

Adobe® is a trademark of Adobe Systems Incorporated.

HP-UX Release 10.20 and later and HP-UX Release 11.00 and later (in both 32 and 64-bit configurations) on all HP 9000 computers are Open Group UNIX 95 branded products.

Intel486 is a U.S. trademark of Intel Corporation.

Java™ is a U.S. trademark of Sun Microsystems, Inc.

Microsoft® is a U.S. registered trademark of Microsoft Corporation.

Netscape™ and Netscape Navigator™ are U.S. trademarks of Netscape Communications Corporation.

OpenView® is a registered U.S. trademark of Hewlett-Packard Company.

Oracle® is a registered U.S. trademark of Oracle Corporation, Redwood City, California.

Oracle Reports™, Oracle7™, and Oracle7 Server™ are trademarks of Oracle Corporation, Redwood City, California.

OSF/Motif® and Open Software Foundation® are trademarks of Open Software Foundation in the U.S. and other countries.

Pentium® is a U.S. registered trademark of Intel Corporation.

SQL*Net® and SQL*Plus® are registered U.S. trademarks of Oracle Corporation, Redwood City, California.

UNIX® is a registered trademark of the Open Group.

Windows NT® is a U.S. registered trademark of Microsoft Corporation.

Windows® 2000 is a U.S. registered trademark of Microsoft Corporation.

Windows® and MS Windows® are U.S. registered trademarks of Microsoft Corporation.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Contents

Contents

Contents

Contents

Contents

Contents

Conventions

The following typographical conventions are used in this manual.

Table 1

Font	What the Font Represents	Example
Italic	Book or manual titles and reference page or manpage names	Refer to the *HP OVW Developer's Guide*.
	Emphasis	You *must* follow these steps.
	A variable that you must supply when entering a command	To open a specific map when starting NNM, type `ovw -map` *`map_name`*, where you supply the map name.
Bold	Terms being defined for the first time	The **distinguishing attribute** of this class...
`Computer`	Text and items on the computer screen	The `Root` map window ... The system prompts: `Press Enter.`
	Cascading menu items	Select `Edit:Find->Object by Comment`
	Command names	Use the `ovstatus` command ...
	File and directory names	`/usr/bin/X11`
	Process names	Check to see if `pmd` is running.
	Window or dialog box names	In the `IP Internet` map window...
`Computer Bold`	Text that you must enter	At the prompt, type: **`ovstatus`**.
Keycap	Keyboard keys	Press **Return**.
`[Button]`	Buttons on the user interface	Click `[NET]`. Click on the `[Apply]` button.

Contact Information

Technical Support and Training

Technical support and training information can be found on the HP OpenView World Wide Web site at:

`http://openview.hp.com/`

Documentation Feedback

Your comments on and suggestions for the documentation help us understand your needs and better meet them.

You can provide feedback about documentation:

- via e-mail to: `ovdoc@fc.hp.com`, or
- via the HP documentation site at: `http://www.docs.hp.com`

If you encounter *serious errors* in the documentation that impair your ability to use the product, please contact the HP Response Center or your support representative so that your feedback can be entered into CHARTS (the HP Change Request Tracking System).

---

1 Overview

SIP Modules to Display NNM Data

HP OpenView Service Information Portal (SIP) includes three modules that display information provided from HP OpenView Network Node Manager (NNM). This book provides information about using these modules, customizing these modules, and even creating your own new-improved versions of these modules. Troubleshooting information is also provided.

Prerequisite: Please read *Configuring NNM* (`Configuring_NNM.pdf`) and follow the instructions for establishing the communication channels between SIP and NNM. The instructions in this book assume that you already completed the steps described in *Configuring NNM*.

What You Can Customize

You can customize any aspect of the modules. This list illustrates some of the possibilities. The chapters in this book provide details about all of your choices:

- Create your own SIP alarm categories by cleverly filtering predefined NNM alarm categories.
- Control the number of alarms to be presented in a given alarm category.
- Create your own network device health gauges to monitor whatever aspects of network health are important to your customers.
- Control how network device health is calculated.
- Change the weights assigned to values in health calculations.
- Change the size of NNM submaps as they appear in SIP portal views.
- Remove status colors from the symbols on the SIP version of NNM submaps, using cream-color rather than the current status color.
- Simplify the amount of information displayed on the submap by perfecting your `MgmtData` filter strategy.
- Substitute bitmaps of NNM submaps instead of gathering them from NNM.

How NNM Works with SIP

SIP can be running on Windows NT/2000, HP-UX, or Solaris and can communicate with multiple NNM management stations and/or collection stations running on any combination of Windows NT/2000, HP-UX, and/or Solaris. For simplicity of terms, NNM management stations and NNM collection stations are both referred to as NNM management stations in SIP documentation. SIP modules provide the ability to aggregate data from multiple NNM sources to display through portal views.

NNM provides up-to-date network status information that you can display to your customers through SIP in the form of alarm lists, topology maps, and/or gauges that measure various aspects of the current network state.

You must first configure NNM and SIP to communicate with each other. See the SIP *Configuring NNM* manual (`Configuring_NNM.pdf`).

How HP OpenView Customer Views Works with SIP

HP OpenView Customer Views runs on top of NNM. If you are using Customer Views, SIP can leverage the customer model that you have already configured, such as assignment of specific devices to a specific organization. See the SIP *Administrator Guide* (`Administrator_Guide.pdf`), “Implementing a Customer Model for Mapping Customers to Resources,” for more information about leveraging the Customer Views configurations to SIP resource mappings.

2 NNM Alarms Module

Understanding NNM Alarms Data

The Alarms module presents network alarms from HP OpenView Network Node Manager (NNM) running on one or more NNM management stations within your management domain.

You control how NNM alarms are displayed within SIP portal views and from which NNM management stations the alarms are gathered by configuring the following files:

- `/registration/OVRegAlarms.xml`
  Registers the Alarms module so that SIP has access. Specifies various properties required by SIP. (See the `OVModuleRegistration.dtd` file for more information.)
- `/registration/defaults/OVDefaultAlarms.xml`
  This is the default Alarms module instance. This file determines the list of NNM alarm categories that are inserted into a portal view when using the `[Add]` button in the SIP user interface. This file is specified in the `defaultConfigXML` attribute in the `OVRegAlarms.xml` file. (See the `/conf/.../views/OVAlarms.dtd` file for more information.)
- `/conf/.../NM/nmConfig.xml`
  This file contains the list of all NNM management stations with which SIP is allowed to communicate. In this file, you specify whether or not the NNM Alarms module is allowed to request data. You also specify which port on the NNM management station SIP needs to contact when requesting alarm data. See `nmConfig.dtd` and `nmConfig.xml` for more information.
- `/conf/.../alarms/NmAlarmsCatIndex.xml`
  Specifies the list of NNM SIP alarm category definitions that are available through SIP. Any alarm categories requested within Alarm module instances, but not listed in this file, are ignored. (See the `/views/OVAlarms.dtd.dtd` file for more information.)
- `/conf/.../alarms/NmAlarmConfig.xml`
  A connection and a thread is established between the SIP server and each NNM station for each active alarm-category/role pair. Multiple portal users viewing alarms that originate from the same NNM management station share a connection as long as they all are assigned to the same role definition. The maximum number of

connections that a SIP server is allowed to establish with NNM management stations (for gathering alarm information) is specified in this file. When the specified maximum is reached, the least used connection is closed and a new one is opened, as needed. (See the `/views/NmAlarmConfig.dtd` file for more information.)

- `/conf/.../alarms/*.xml` (based upon the `nmAlarmCat.dtd`) There is one file for each NNM alarm category that you wish to display in SIP portals. You configure:
  - Title to use within SIP (which can be different from the name of the base alarm category in NNM, and different from the XML filename).
  - Maximum number of alarms to display
  - Which NNM alarm category to use as a starting point
  - Which NNM management stations to gather alarms from (must be empty or a subset of those listed in the `nmConfig.xml` file)
  - How many minutes to wait after NNM receives the alarm before displaying it in the SIP portal
  - Elements and attributes that filter alarms prior to displaying them in the SIP portal (for example: by specific alarm severity settings, acknowledged and/or unacknowledged, alarms containing a specific text string in the description field).
- Additional filter information
  The alarms are automatically filtered according to the `MgmtData` filter assigned to the customer logging into the specific portal view. To further restrict data, you can write a `NodeSelection` filter within each `/alarms/alarmCatetgory.xml` (see `/views/filter.dtd` for more information). The following are allowed within NNM alarm category `NodeSelection` filter:
  - `IPHostFilter` and/or
  - `CapabilityFilter` and/or
  - `OrganizationFilter`

- `/views/PortalView.xml`
  Place the actual instances of the NNM Alarms module `<Summary>` within the `PortalView.xml` file. Specify which SIP alarm categories `<CategoryDefName href="nmAlarmCat.xml"/>` are visible in the portal view for each instance. (See the `/conf/.../views/OVAlarms.dtd` file for more information.)

The following topics are covered in the remainder of this chapter:

“Registering the Alarms Module” on page 23

“Filtering Possibilities for the Alarms Module” on page 25

“Establishing Global Settings for Alarms Modules” on page 28

“Editing Alarm Categories” on page 31

“Editing the NNM Alarms Module” on page 40

“Location of Relevant Files” on page 46

Registering the Alarms Module

The Alarms module must be registered with SIP.

This section focuses on the following two files:

- `/registration/OVRegAlarms.xml`
- `/registration/defaults/OVDefaultAlarms.xml`

See the `OVModuleRegistration.dtd` file and the SIP *Administrator Guide* (`Administrator_Guide.pdf`) for more information.

Editing Access

The following attribute in the `OVRegAlarms.xml` file is automatically enabled or disabled based upon the current SIP portal user's Role *editing permissions*:

- `edit`
  This attribute is set to *yes*. When a user with "Editing" permissions displays a Alarms module in the SIP portal, an edit button appears in the titlebar of the Alarms module. This button provides access to limited editing functions: display alarm categories, remove alarm categories, change the order of alarm categories.

For more information about roles and editing permissions, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`).

The [Add] Button

The following attributes in the `OVRegAlarms.xml` file are automatically enabled or disabled based upon the current SIP portal user's Role *editing permissions*:

- `add`
  Set this attribute to *yes*. The default Alarms module appears in the list of available modules when a user with "Editing" permissions uses the [Add] button at the bottom of the SIP portal window.
- `defaultConfigXML`
  This attribute specifies which Alarms module appears when a user with "Editing" permissions uses the [Add] button in a SIP portal to

insert an Alarms module. The `OVDefaultAlarms.xml`file contains the default Alarms module. Specify any file that you want, or modify the Alarms module within the current file to meet your needs. See the comments in the `/conf/.../views/OVAlarms.dtd` file for more information.

For more information about roles and editing permissions, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`).

The [?] (Help) Button

The following attribute in the `OVRegAlarms.xml` file controls the default behavior of the `[?]` button in the Alarms module titlebar:

- `help`
  This attribute specifies the default help topic (html file) that appears when a user clicks the `[?]` button on an Alarms module titlebar: `/OvSipDocs/C/help/NNM/alarmsView.html`. You can specify any html file that you want.

You can override the default topic and provide a customized topic on a module-by-module basis.

TIP

To specify a more specific help file on a module-by-module basis, add a `help` attribute to the `ModuleInstance` element. For more information, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`), "Creating Customized Help Topics for Supplied Modules."

Filtering Possibilities for the Alarms Module

The Effect of MgmtData Filter and Roles

Data displayed in the Alarms module must pass the `MgmtData` filter you defined in the customer model configuration for each role. See the "Filtering Data by Customer" in the SIP *Administrator Guide* (`Administrator_Guide.pdf`)for information about the `MgmtData` filter, customer model, and roles.

In addition to the filtering you applied through the `MgmtData` filter, a finer, second level of filtering is available by applying a `NodeSelection` filter to each alarm category. A variety of other filtering attributes and elements are included in each alarm category definition.

Node Selection Filters for Specific Alarm Categories

Whereas the `MgmtData` filter defines what the portal view can *potentially* display, the `NodeSelection` filter provides a finer level of control over what the portal view *actually* displays. The `NodeSelection` filter is defined for individual alarm categories to further restrict the set of nodes whose alarms are displayed.

The `NodeSelection` filter has three child elements that further restrict the alarms allowed to pass:

- `IPHostFilter`
  Passes only alarms from devices specified by hostname or IP address.
- `CapabilityFilter`
  Passes only alarms from devices having the specified `capability` attribute within the NNM database, for example, `isRouter`.
- `OrganizationFilter`
  Passes only alarms from devices included in the specified customer model *organization* element.

When writing filters, use Perl5 regular expressions. Periods in entries must be escaped with a backslash (\) character and the asterisks (*) must be escaped with a period (.): for example:
`.*\.eagle\.wingnuts\.com`

The actual nodes whose alarms are displayed result from the *intersection* of nodes between the `MgmtData` filter and the `NodeSelection` filter. It is possible at runtime for the *intersection* of these lists to be the empty set. In this case, no nodes are selected and no output occurs for this alarm category.

Leaving the `NodeSelection` filter empty results in all nodes that pass the `MgmtData` filter passing the `NodeSelection` filter when determining the intersection of candidate nodes.

For more information about implementing the `NodeSelection` filter, see the following:

`/conf/share/views/filter.dtd`
"Filtering Data by Customers" section of the SIP *Administrator Guide* (`Administrator_Guide.pfd`)

Additional Filtering Attributes and Elements

All of the choices explained below are specified in the each alarm category definition file (*`NmAlarmCat`*`.xml`). See `NmAlarmConfig.dtd` and `SampleNmAlarmCat.xml` for more information.

Alarms that pass the `MgmtData` filter and `NodeSelection` filter are further filtered according to the following criteria as an AND condition before being displayed in any SIP portal view:

- `NNMStationList`
  (optional, subset of NNM management stations listed in the `nmConfig.xml` file. Alarms for this category are gathered only from the specified NNM management stations. If empty, all stations specified in the `nmConfig.xml` file pass.)

NOTE Verify that you are consistent in your usage of *either* hostname or IP address when specifying the NNM management station in the `nmConfig.xml` file and any alarm category definition files (*`NmAlarmCat`*`.xml`files).

- `NNMBaseCategory`
  (required, the alarm category currently defined in NNM that you wish to access for this SIP alarm category)

- `MatchDescSubstring`
  (optional, display only alarms whose messages include the specified text. If empty all descriptions pass.)
- `OlderThanXMinutes`
  (optional, wait the specified number of minutes before displaying any alarm. If empty, show alarms as soon as they happen.)
- `Severities`
  (required, the NNM-defined alarm severity levels that you wish to include. No alarms pass if you specify none.)
- `Acknowledgement`
  (required, include alarms that are acknowledged and/or unacknowledged within NNM. No alarms pass if you specify neither.)

For example, the alarm criteria may specify NodeSelection:`host.corp.com` and Severity: `critical`. In this example, only alarms with the source name matching `host.corp.com` AND having a severity of `critical` are displayed.

If any of the above support multiple values, each value is treated as an OR condition. To continue the example, with severities `critical` and `major`, alarms matching the hostname `host.corp.com` *AND* having either a severity of `critical` *OR* `major` are displayed. If multiple nodes are supplied in the node list (such as `hostA.corp.com;hostB.corp.com`) alarms matching EITHER `hostA` OR `hostB` are displayed.

Establishing Global Settings for Alarms Modules

This section focuses on the following two files:

- `/conf/share/modules/alarms/NmAlarmConfig.xml`
- `/conf/share/modules/NM/nmConfig.xml`

If you make changes to either of these files, you must do the following:

WindowsNT/2000:
From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

UNIX:

As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

Stop on HP-UX: `/sbin/init.d/ovsip stop`
Start on HP-UX: `/sbin/init.d/ovsip start`

Stop on Solaris: `/etc/init.d/ovsip stop`
Start on Solaris: `/etc/init.d/ovsip start`

NmAlarmConfig.xml/dtd

The attributes defined in this file affect all alarm categories in every instance of an Alarms module in all *`PortalView`*`.xml` files:

- `maxConnections`
  The maximum number of (socket) connections that a SIP server is allowed to establish with all the NNM management stations it needs to communicate with, for gathering alarm information. A socket connection opens for each alarm category, for each role, for each NNM management station.

  For example, an NNM Alarms module that has 6 alarm categories and gathers alarms from 6 NNM management would require 36 socket connections per role. If there are 2 roles, 78 socket connections

are required. Multiple users can share the same role, and therefore share socket connections. When the specified number is reached, for each new subsequent connection required: (1) the least used connection is closed and (2) a new one is opened.

- `showSummaryLine`
  yes means a message displays at the bottom of each alarm category explaining current configuration settings ("configured for x alarms, received y.")
  no means no such message is displayed.
- `shortDateFormat`
  yes means the current locale setting's short date format is used. For example, US English: `mm/dd/yy hh:mm:ss am/pm`
  no means the current locale's long date format is used. For example, US English: `Tuesday March 20 2001 hh:mm:ss am/pm tz`
- `connTimeOut` (zero or greater)
  The number of seconds to pause after each socket connection is opened. The smaller the number, the faster the Alarms module opens when a *`PortalView`*`.xml` is first accessed. However, the `connTimeout` value may be so short that no alarms are displayed until the portal is refreshed. Too short a `connTimeOut` value causes the "Data currently unavailable" error message to display, rather than the alarm text.
- `addSyncTime` (zero or greater)
  The number of seconds to add to `connTimeout` when making a synchronous call to get data from the `ovalarmsrv` on each NNM management station. Synchronous calls are required when the you set the `OlderThanXMinutes` attribute to a non-zero value in any *`NmAlarmCat`*`.xml` file. Once an `OlderThanXMinutes` attribute is specified, alarm data cannot be cached because the time value in the filter request changes with every refresh.
- `socketTimeout` (zero or greater)
  The number of seconds to wait for a socket connection to be made.
- `responseTimeout` (zero or greater)
  The number of seconds to wait each time for any response (protocol or data) from `ovalarmsrv`.

- `maxWaitTime` (zero or greater)
  The maximum number of seconds to wait for a data response from `ovalarmsrv`. The value for this attribute should be greater than the value for the `responseTimeout` attribute to allow for delays due to network traffic.

See the `NmAlarmConfig.dtd` and `NmAlarmConfig.xml` for more information.

nmConfig.xml/dtd

Only those NNM management stations or collection stations that are configured to do so in the `nmConfig.xml` file are allowed to provide alarm information to SIP. See `nmConfig.xml` and `nmConfig.dtd` for more information. See also the `Configuring_NNM.pdf` file. The bold settings in the following example must be set before the Alarms module works:

```
<NNMStation
     hostname="hostname or IP address of the NNM system"
     snmpDataSource="get SNMP data collection data?: yes or no"
     alarmsDataSource="get NNM alarms?: yes or no"
     symbolRegSource="get OVW symbol registration information?: yes or no"
     webSrvPort="NNM web server port: usually 80 for NT; 8880 for Unix"
     ovwdbPort="ovwdb port: usually 9999 for NNM 6.1; 2447 for NNM 6.2"
     ovAlarmSrvPort="ovalarmsrv port: 2345 for NNM 6.1; 2953 for NNM 6.2"
     encoding="if NNM is using other than English"
/>
```

The `alarmsDataSource` attribute controls which NNM management stations are polled for current NNM alarms.

When the NNM management station is running in a language other than English, see the "Internationalization" chapter of the SIP *Administrator Guide* (`Administrator_Guide.pdf`) for additional instructions.

NOTE Verify that you are consistent in your usage of *either* hostname or IP address when specifying the NNM management station in the `nmConfig.xml` file and any alarm category definition files (*`NmAlarmCat`*`.xml`files).

Editing Alarm Categories

SIP alarm categories are based upon existing NNM alarm categories, such as “Status Alarms” or “Threshold Alarms.” SIP alarm categories must be defined in an *NmAlarmCat*.xml file and be listed in the *NmAlarmCatsIndex*.xml file before they can be displayed in SIP portal views through Alarms modules. Although they are based upon a specific NNM alarm category, the SIP alarm category name can be different from the base NNM alarm category name; such as “Accounting Department’s Network Problems” or “Internet Availability Alarms.”

You can modify SIP alarm categories in a variety of ways:

“Modifying Existing Alarm Categories” on page 31

“Specifying the Number of Alarms to Display” on page 33

“Creating New Alarm Categories” on page 35

“Eliminating an Alarm Category” on page 38

Modifying Existing Alarm Categories

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. Before you start, access the following two files as a reference:
   - *Windows NT/2000:*
     <SIP_install_dir>\SIP\conf\share\modules\alarms\SampleNmAlarmCat.xml and NmAlarmCat.dtd
   - *UNIX:*
     /etc/opt/OV/SIP/conf/share/modules/alarms/SampleNmAlarmCat.xml and NmAlarmCat.dtd
2. In an ASCII or XML editor, open the *NmAlarmCat*.xml file that defines the alarm category you wish to modify. These definition files must be stored in the following location:

- *Windows NT/2000:*
  <SIP_install_dir>\SIP\conf\share\modules\alarms\
- *UNIX:*
  /etc/opt/OV/SIP/conf/share/modules/alarms/

3. Following the rules as explained in the SampleNmAlarmCat.xml and NmAlarmCat.dtd files, make the desired modifications.

   The *NmAlarmCat*.xml file controls the filtering specifications for the alarm category (as explained in "Node Selection Filters for Specific Alarm Categories" on page 25 and "Additional Filtering Attributes and Elements" on page 26).

   The *NmAlarmCat*.xml file also controls the displayed SIP alarm category title and the maximum number of alarms that can be displayed within a SIP portal view.

NOTE Verify that you are consistent in your usage of *either* hostname or IP address when specifying the NNM management station in the nmConfig.xml file and any alarm category definition files (*NmAlarmCat*.xmlfiles).

4. Close and save the *NmAlarmCat*.xml file.
5. *WindowsNT/2000*:
   From the Control Panel, select Services. Stop and then restart Tomcat. Alternatively, you can use the command line: **net stop tomcat** and **net start tomcat**.

   UNIX:

   As root, stop and restart the web server and servlet engine by running the following. (The DISPLAY variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **/sbin/init.d/ovsip stop**
   Start on HP-UX: **/sbin/init.d/ovsip start**

   Stop on Solaris: **/etc/init.d/ovsip stop**
   Start on Solaris: **/etc/init.d/ovsip start**
6. In a browser, log into the portal to verify that the alarms appear as desired.

You may wish to review the current global alarm category settings (see "Establishing Global Settings for Alarms Modules" on page 28).

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Specifying the Number of Alarms to Display

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *`NmAlarmCat`*`.xml` file that defines the alarm category you wish to modify. These definition files must be stored in the following location:

   - *Windows NT/2000:*
     `<`*`SIP_install_dir`*`>\SIP\conf\share\modules\alarms\`
   - *UNIX:*
     `/etc/opt/OV/SIP/conf/share/modules/alarms/`

2. Locate the `NumAlarms` attribute and change the value to the number of alarms you wish to display in this SIP alarm category through all

portal views.

3. Close and save the *NmAlarmCat*.xml file.
4. *WindowsNT/2000*:
   From the Control Panel, select Services. Stop and then restart Tomcat. Alternatively, you can use the command line: **net stop tomcat** and **net start tomcat**.

   UNIX:

   As root, stop and restart the web server and servlet engine by running the following. (The DISPLAY variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **/sbin/init.d/ovsip stop**
   Start on HP-UX: **/sbin/init.d/ovsip start**

   Stop on Solaris: **/etc/init.d/ovsip stop**
   Start on Solaris: **/etc/init.d/ovsip start**
5. In a browser, log into the portal to verify that the alarms appear as desired.

You may wish to review the current global alarm category settings (see "Establishing Global Settings for Alarms Modules" on page 28). One of the global choices specifies whether or not a message is displayed, at the bottom of each alarm category, explaining the number of alarms currently allowed to be displayed.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command xmlvalidate, which checks whether the XML file is both well-formed and valid. xmlvalidate uses the same XML parser as SIP, so if the file passes xmlvalidate, it will work with SIP. The correct usage of this command is: xmlvalidate <xml filename>.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: %SIP_HOME%\bin
UNIX: /opt/OV/SIP/bin

If the output of the xmlvalidate command indicates a problem but does

not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Creating New Alarm Categories

Each SIP alarm category is defined in a separate XML file.

When configuring a new alarm category, first decide which NNM Alarms Category you want to use in a given portal view and which NNM management stations to collect alarms from. SIP ships with predefined alarm category files for the standard NNM alarm categories. You can copy any of these to use as a starting point.

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. Before you start, access the following two files as a reference:
   - *Windows NT/2000:* `<SIP_install_dir>\SIP\conf\share\modules\alarms\SampleNmAlarmCat.xml` and `NmAlarmCat.dtd`
   - *UNIX:* `/etc/opt/OV/SIP/conf/share/modules/alarms/SampleNmAlarmCat.xml` and `NmAlarmCat.dtd`
2. Copy and rename (*`NmAlarmCat`*`.xml`) one of the following as a starting point for your new alarm category definition file:
   - `SampleNmAlarmCat.xml`
   - `ApplicationAlertAlarms.xml`
   - `ConfigurationAlarms.xml`
   - `ErrorAlarms.xml`
   - `StatusAlarms.xml`
   - `ThresholdAlarms.xml`

3. In an ASCII or XML editor, open the *NmAlarmCat*.xml file that defines the alarm category you wish to modify. This file must be stored in the following location:

   - *Windows NT/2000:*
     <SIP_install_dir>\SIP\conf\share\modules\alarms\
   - *UNIX:*
     /etc/opt/OV/SIP/conf/share/modules/alarms/

4. Following the rules as explained in the SampleNmAlarmCat.xml and NmAlarmCat.dtd files, make the desired modifications.

   The *NmAlarmCat*.xml file controls the filtering specifications for the alarm category (as explained in "Node Selection Filters for Specific Alarm Categories" on page 25 and "Additional Filtering Attributes and Elements" on page 26).

   The *NmAlarmCat*.xml file also controls the displayed SIP alarm category title and the maximum number of alarms that can be displayed within a SIP portal view.

CAUTION

If you are gathering alarms from an NNM management station running in a language other than English, see the SIP *Administrator Guide* (Administrator_Guide.pdf) for important information about localization of the Alarms module.

NOTE

Verify that you are consistent in your usage of *either* hostname or IP address when specifying the NNM management station in the nmConfig.xml file and any alarm category definition files (*NmAlarmCat*.xmlfiles).

5. Close and save the *NmAlarmCat*.xml file.
6. In an ASCII or XML editor, open the NmAlarmsCatIndex.xml file that contains the list of all valid alarm categories available for display in your portal view. This file must be stored in the following location:

   - *Windows NT/2000:*
     <SIP_install_dir>\SIP\conf\share\modules\alarms\

- *UNIX:*
  `/etc/opt/OV/SIP/conf/share/modules/alarms/`

7. *WindowsNT/2000*:
   From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

   UNIX:

   As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **`/sbin/init.d/ovsip stop`**
   Start on HP-UX: **`/sbin/init.d/ovsip start`**

   Stop on Solaris: **`/etc/init.d/ovsip stop`**
   Start on Solaris: **`/etc/init.d/ovsip start`**

8. In a browser, log into the portal to verify that the alarms appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Eliminating an Alarm Category

1. In an ASCII or XML editor, open the *NmAlarmsCatsIndex*.xml file:
   - *Windows NT/2000:*
     <*SIP_install_dir*>\SIP\conf\share\modules\alarms\
   - *UNIX:*
     /etc/opt/OV/SIP/conf/share/modules/alarms/
2. Delete the CategoryDefName line referring to the SIP alarm category that you wish to eliminate.
3. Close and save the *NmAlarmsCatsIndex*.xml.xml file.
4. *WindowsNT/2000*:
   From the Control Panel, select Services. Stop and then restart Tomcat. Alternatively, you can use the command line: **net stop tomcat** and **net start tomcat**.

   UNIX:

   As root, stop and restart the web server and servlet engine by running the following. (The DISPLAY variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **/sbin/init.d/ovsip stop**
   Start on HP-UX: **/sbin/init.d/ovsip start**

   Stop on Solaris: **/etc/init.d/ovsip stop**
   Start on Solaris: **/etc/init.d/ovsip start**
5. In a browser, log into the portal to verify that the alarms appear as desired.

This alarm category is no longer allowed to display in SIP portal views, even if it is specifically listed in an Alarms module within a PortalView.xml file.

If you want to totally eliminate any trace of this SIP alarm category:

1. In an ASCII or XML editor, open the following files and delete all instances of the CategoryDefName line referring to the obsolete SIP alarm category and save the modified version of the files:

- *Windows NT/2000:*

```
<SIP_install_dir>\SIP\conf\share\views\*.xml
<SIP_install_dir>\SIP\registration\defaults\OVDefaultAlarms.xml
```

- *UNIX:*

```
/etc/opt/OV/SIP/conf/share/views/*.xml
/etc/opt/OV/SIP/registration/defaults/OVDevaultAlarms.xml
```

2. Delete the *NmAlarmCat*.xml file that defines the obsolete alarm category. These definition files must be stored in the following location:

   - *Windows NT/2000:*
     <SIP_install_dir>\SIP\conf\share\modules\alarms\
   - *UNIX:*
     /etc/opt/OV/SIP/conf/share/modules/alarms/

3. *WindowsNT/2000*:
   From the Control Panel, select Services. Stop and then restart Tomcat. Alternatively, you can use the command line: **net stop tomcat** and **net start tomcat**.

   UNIX:

   As root, stop and restart the web server and servlet engine by running the following. (The DISPLAY variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **/sbin/init.d/ovsip stop**
   Start on HP-UX: **/sbin/init.d/ovsip start**

   Stop on Solaris: **/etc/init.d/ovsip stop**
   Start on Solaris: **/etc/init.d/ovsip start**

Editing the NNM Alarms Module

You control which SIP Alarm Categories are visible:

- Adding an Alarms Module to a Portal View
- Choosing Alarm Categories in an Alarms Module
- Removing Alarm Categories from an Alarms Module
- Changing the Display Order of Alarm Categories

Adding an Alarms Module to a Portal View

Two ways exist for adding Alarms modules to a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *PortalView*.xml file you want to customize.
2. Navigate to the appropriate Tab.
3. At the bottom of the window, select Alarms and click [Add].
4. Save the changes and return to the main portal page; click the [OK] button.

TIP If you want to create and add a different instance of a module to the list of available modules, see the SIP *Administrator Guide* (Administrator_Guide.pdf)

Through the Portal View File

Modules can be added and configured by directly editing a PortalView.xml file. Modules are wrapped in the ModuleInstance element. The ModuleInstance id must be unique among all module instances in the portal view file. For information about the ModuleInstance element, see the SIP *Administrator Guide*

(Administrator_Guide.pdf), “Designing Portal Views.”

Follow the directions in the SIP *Administrator Guide*.

Refer to the OVAlarms.dtd file for more information.

You can copy and paste the contents of either of the following files into your *PortalView*.xml file as a starting point:

- NmAlarmCatsIndex.xml
- OVDefaultAlarms.xml

Choosing Alarm Categories in an Alarms Module

Two ways exist for choosing the alarm categories presented in a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *PortalView*.xml file you want to customize.
2. Navigate to the appropriate Tab.

   If necessary, navigate to the [Add] button. Select Alarms and click [Add].
3. On the title bar of the Alarms module, click the [Edit] button.
4. On the Alarms - Edit page, select an alarm category in the Available Alarm Categories list, and click the [Add] button.
5. Repeat step 4 until the Displayed Alarm Categories list contains all categories you want to display.
6. Save the changes and return to the main portal page; click the [OK] button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the `NmAlarmsCatIndex.xml` file that contains the list of all valid alarm categories available for display in your portal view:

   - *Windows NT/2000:*<

`<SIP_install_dir>\conf\share\modules\alarms\NmAlarmsCatIndex.xml`

   - *UNIX:*

`/etc/opt/OV/SIP/conf/share/modules/alarms/NmAlarmsCatIndex.xml`

2. Copy all `CategoryDefName` lines.
3. In an ASCII or XML editor, open the *`PortalView`*`.xml` where you wish to add an alarm category to the Alarms module.
4. Navigate to the appropriate Alarms module, and paste the `CategoryDefName` lines between the `<AlarmDisplay>` and `</AlarmDisplay>` elements.
5. Delete any `CategoryDefName` lines that you do not wish to display.
6. Rearrange the `CategoryDefName` lines appropriately.
7. Close and save the *`PortalView`*`.xml` file.
8. In a browser, log into the portal to verify that the alarms appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Removing Alarm Categories from an Alarms Module

Two ways exist for removing alarm categories from a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *`PortalView`*`.xml` file you want to customize.
2. Navigate to the appropriate Tab.
3. On the title bar of the Alarms module, click the `[Edit]` button.
4. On the `Alarms - Edit` page, select an alarm category in the `Displayed Alarm Categories` list, and click the `[Remove]` button.
5. Repeat step 4 until the `Displayed Alarm Categories` list contains only the alarm categories you want to display.
6. Save the changes and return to the main portal page; click the `[OK]` button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *`PortalView`*`.xml` where you wish to remove an alarm category from the Alarms module.
2. Navigate to the appropriate Alarms module.
3. Delete any `CategoryDefName` lines that you do not wish to display.
4. Close and save the *`PortalView`*`.xml` file.
5. In a browser, log into the portal to verify that the alarms appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Changing the Display Order of Alarm Categories

Two ways exist for choosing the display order of alarm categories in a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *`PortalView`*`.xml` file you want to customize.
2. Navigate to the appropriate Tab.
3. On the title bar of the Alarms module, click the `[Edit]` button.
4. On the `Alarms - Edit` page, select an alarm category in the `Displayed Alarm Categories` list and click the `[Up]` or `[Down]` button.
5. Repeat step 4 until the `Displayed Alarm Categories` list displays the categories in the order you prefer.

6. Save the changes and return to the main portal page; click the [OK] button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *PortalView*.xml where you wish to add an alarm category to the Alarms module.
2. Navigate to the appropriate Alarms module.
3. Rearrange the CategoryDefName lines appropriately.
4. Close and save the *PortalView*.xml file.
5. In a browser, log into the portal to verify that the alarms appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command xmlvalidate, which checks whether the XML file is both well-formed and valid. xmlvalidate uses the same XML parser as SIP, so if the file passes xmlvalidate, it will work with SIP. The correct usage of this command is: xmlvalidate <xml filename>.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: %SIP_HOME%\bin
UNIX: /opt/OV/SIP/bin

If the output of the xmlvalidate command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at www.xmlspy.com.

Location of Relevant Files

Table 2-1 Alarms Module Files on the SIP Server

File Name	Windows NT/2000 Location <*SIP_install_dir*>/SIP/....	UNIX Location etc/opt/OV/SIP/....
nmConfig.dtd	conf/share/modules/NM/	conf/share/modules/NM/
nmConfig.xml	conf/share/modules/NM/	conf/share/modules/NM/
OVModuleRegistration.dtd	registration/	registration/
OVRegAlarms.xml	registration/	registration/
NmAlarmCat.dtd	conf/share/modules/alarms/	conf/share/modules/alarms/
NmAlarmCat.xml	conf/share/modules/alarms/	conf/share/modules/alarms/
NmAlarmConfig.dtd	conf/share/modules/alarms/	conf/share/modules/alarms/
NmAlarmConfig.xml	conf/share/modules/alarms/	conf/share/modules/alarms/
OVAlarms.dtd	conf/share/views/	conf/share/views/
OVDefaultAlarms.xml	registration/defaults/	registration/defaults/
NmAlarmCatsIndex.dtd	conf/share/modules/alarms/	conf/share/modules/alarms/
PortalView.xml	conf/share/views	conf/share/views

3 Network Device Health Gauge Module

Understanding Network Device Health Gauges

Network health is scored as a value from 0-100, with 0 being the poorest health and 100 being the best health.

Two types of views are available for Network Device Health: Gauges and Details. Gauges are displayed when the tab first displays in your portal, and indicate the overall health rating for all devices being monitored by the particular gauge. Details are viewed by clicking on a gauge or a health title link.

Gauge View

You can customize the health gauges so that your customers see only the gauges you want them to see. You can modify the predefined gauges or write your own to monitor any aspect of the network that is of concern to your customers.

A network health gauge represents the mean health of all network devices being monitored by a particular gauge. For example, `Router Health` represents the mean health of all routers. If you have two routers, one with a health score of 100% and one with a score of 60%, Router Health points to 80%.

The predefined gauges that ship with SIP are:

- CPE (Customer Premises Equipment) Health Gauge
- Interface Health Gauge
- Key Device Health Gauge
- Server Health Gauge
- Router Health Gauge

Details View

Health gauges may have two levels of health detail drill-down. If available:

- To view the first level health details table, click on the gauge or the health gauge title above the gauge.

- To view the second level of health detail, click the health score values that are hyperlinks in the first level health detail table. Only certain values provide hyperlinks.

The first column of a detail page—**Resource**—displays the name of the network resource (for example, the name of the Interface, Router, Server, Key Device, or Customer Premises Equipment).

The second column—**Overall Health**—contains the resource's health score. This score is based upon the weighted mean of a set of statistics measured on that resource.

The remaining columns display the health score for each statistic used to compute network resource health. The score is a value from 0-100 derived from analysis of the Metric value. You choose whether or not to present raw data in your portal views. By default, raw data is not presented.

The algorithms for health analysis are defined in the `netHealthConfig.dtd` file and `netHealthConfig.xml` file:

Windows NT/2000:
`<SIP_install_dir>\conf\share\modules\health\`

UNIX: `/etc/opt/OV/SIP/conf/share/modules/health/`

The tables below describe the default statistics used for the default health categories.

By default, health of a single interface is derived using the four statistics described in the Table 3-1, "Interface Health Gauge." Health of all interfaces on a Router, Key Device, CPE, or Server is also derived from those four statistics.

Table 3-1 Interface Health Gauge

Statistic	Default Settings Description
Up/Down Status	An indication of whether the interface is up or down. An interface that is up has a status health score of 100%. An interface that is down has a status health score of 0%. Because this is an important measure of health, status is given double the weight (by default) than the other statistics when overall interface health is computed.
Utilization Health	The percent utilization of an interface. For example, a 50% utilization health score means that NNM measured the available bandwidth on an interface, and found that 50% was being used. Higher utilization rates translate into lower utilization health scores.
Inbound Error Health	The error rate (percent) for inbound data on the interface. High error rates translate into lower inbound error health scores.
Outbound Error Health	The error rate (percent) for outbound data on the interface. High error rates translate into lower outbound error health scores.

Table 3-2 Router Health Gauge

Statistic	Default Settings Description
Interface Health	The mean of the health scores for all active interfaces within the routers. (Active interfaces are those that are not administratively down and are not loopback type interfaces.) The health score of each interface is calculated using the four components explained in the Interface Health Gauge table. Clicking on an Interface Health Score accesses the detail health information for the interfaces for that router.
CPU Utilization Health	The percent utilization of the router's CPU. For example, a 50% utilization health score means that NNM measured the available CPU bandwidth, and found that 50% was being used. High utilization metric values translate into low utilization health scores.

NOTE For interface metrics, different formulas are used depending upon the attributes of the interface (such as speed of the interface, half-duplex versus full-duplex, etc.). In the case of CPU utilization, the expression is really just a single Cisco MIB object: `local.system.augBusy5`. It is described as the "5 minute exponentially-decayed moving average of the CPU busy percentage."

Table 3-3 Key Device Health, CPE Health, and Server Health Gauges

Statistic	Default Settings Description
Interface Health	The mean of the health scores for all active interfaces within the specified network devices. (Active interfaces are those that are not administratively down and are not loopback type interfaces.) The health score of each interface is calculated using the four components explained in the Interface Health Gauge table. Clicking on an Interface Health Score accesses the detail health information for the interfaces for that device.

Overview of Files Involved

You control how Network Device Health gauges are displayed within SIP portal views and from which NNM management stations the data is gathered by configuring the following files:

- `/registration/OVRegNetHealth.xml`
  Registers the Network Device Health module so that SIP has access. Specifies various properties required by SIP. (See the `OVModuleRegistration.dtd` file for more information.)
- `/registration/defaults/OVDefaultNetHealth.xml`
  Defines the default Network Device Health module instance. This module is inserted into a portal view when using the `[Add]` button in the SIP user interface. This file is specified in the `defaultConfigXML` attribute in the `OVRegNetHealth.xml` file. (See the `/conf/.../views/OVNetworkHealth.dtd` file for more information.)
- `/conf/.../NM/nmConfig.xml`
  This file contains the list of all NNM management stations with which SIP is allowed to communicate. You must specify whether or not the Network Device Health module should retrieve SNMP data. You provide information about which port to connect to when collecting health data. See `nmConfig.dtd` and `nmConfig.xml` for more information.

- `/conf/.../health/netHealthConfig.xml`
  This file contains configuration information that applies, or potentially applies, to all health gauges for all portal users. It also contains a `Metric` element for each MIB object and each MIB expression used by any gauge. The `Metric` entries are reusable in multiple gauges. The `Metric` determines the rules for calculating and displaying the `score` associated with each returned MIB value. (See the `netHealthConfig.dtd` file for more information.)

- Filter information
  The gauges are automatically filtered according to the `MgmtData` filter assigned to the role of the customer logging into the portal view. To further restrict data, you write a `NodeSelection` filter and/or an `InterfaceSelection` filter within the Network Device Health gauge definition. See “Filtering Possibilities for the Network Device Health Module” on page 57, the `/views/filter.dtd` file, and "Filtering Data by Customer" in the SIP *Administrator Guide* (`Administrator_Guide.pdf`) for more information about writing filters.

  The `NodeSelection` filter contains any combination of the following:

  — `IPHostFilter`

  — `CapabilityFilter`

  — `OrganizationFilter`

  The `InterfaceSelection` filter contains:

  — `IPInterfaceFilter`

  — `OrganizationFilter`

- `/views/`*`PortalView`*`.xml`
  These are the files that define the portals that your customers see. Place the actual instance of the Network Device Health module `Summary` within the *`PortalView.xml`* file. Specify which gauges are visible in the portal view for each module instance. (See the `/conf/.../views/OVNetworkHealth.dtd` file for more information.)

NOTE Multiple files are involved in the actual data collection process. See "Collecting Data for Network Device Health Gauges" on page 88 and the *Configuring NNM* manual (`Configuring_NNM.pdf`) for more information.

The following topics are covered in the remainder of this chapter:

Registering the Network Device Health Module

The Network Device Health module must be registered with SIP.

This section focuses on the following two files:

- `/registration/OVRegNetHealth.xml`
- `/registration/defaults/OVDefaultNetHealth.xml`

See the `OVModuleRegistration.dtd` file and the SIP *Administrator Guide* (`Administrator_Guide.pdf`) for more information.

Editing Access

The following attribute in the `OVRegNetHealth.xml` file is automatically enabled or disabled based upon the current SIP portal user's Role *editing permissions*:

- `edit`
  This attribute is set to *yes*. When a user with "Editing" permissions displays a Network Device Health module in the SIP portal, an edit button appears in the titlebar of the Network Device Health module. This button provides access to limited editing functions: display a gauge, remove a gauge, change the order of gauges.

For more information about roles and editing permissions, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`).

The [Add] Button

The following attributes in the `OVRegNetHealth.xml` file are automatically enabled or disabled based upon the current SIP portal user's Role *editing permissions*:

- `add`
  Set this attribute to *yes*. The default Network Device Health module appears in the list of available modules when a user with "Editing" permissions uses the [Add] button at the bottom of the SIP portal window.

- `defaultConfigXML`
  This attribute specifies which Network Device Health module appears when a user with "Editing" permissions uses the `[Add]` button in a SIP portal to insert a Network Device Health module. The `OVDefaultNetHealth.xml`file contains the default Network Device Health module. Specify any file that you want, or modify the Network Device Health module within the current file to meet your needs. See the comments in the `/conf/.../views/OVNetHealth.dtd` file for more information.

For more information about roles and editing permissions, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`).

The [?] (Help) Button

The following attribute in the `OVRegNetHealth.xml` file controls the default behavior of the `[?]` button in the SIP portal:

- `help`
  This attribute specifies the default help topic (html file) that appears when a user clicks the `[?]` button on a Network Device Health module: `/OvSipDocs/C/help/NNM/healthView.html`. You can specify any html file that you want.

You can override the default topic and provide a customized topic on a module-by-module basis.

TIP To specify a more specific help file on a module-by-module basis, add a `help` attribute to the `ModuleInstance` element. For more information, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`), "Creating Customized Help Topics for Supplied Modules."

Filtering Possibilities for the Network Device Health Module

The Effect of MgmtData Filter and Roles

Data displayed in the Network Device Health module must pass the `MgmtData` filter you defined in the customer model configuration for each role. See "Filtering Data by Customer" in the SIP *Administrator Guide* (`Administrator_Guide.pdf`)for information about the `MgmtData` filter, customer model, and roles.

In addition to the filtering you applied through the `MgmtData` filter, a finer, second level of filtering is applied with a `NodeSelection` filter and/or `InterfaceSelection` filter in each gauge. For more information about filtering, see:

- `OVNetworkHealth.dtd`
- `OVDefaultNetHealth.xml`
- `filter.dtd`

NodeSelection and InterfaceSelection Filters for Specific Health Gauges

Whereas the `MgmtData` filter defines what the portal view can *potentially* display, the `NodeSelection` filter and `InterfaceSelection` filter provide a finer level of control over what the portal view *actually* displays. These filters are defined for individual health gauges to further restrict the set of nodes and interfaces whose health is included in the calculations. It is required that either the `NodeSelection` filter or `InterfaceSelection` filter be specified. Both can be specified in the same gauge, if desired.

The `NodeSelection` filter has three potential child elements that further restrict the nodes allowed to pass:

- `IPHostFilter`
  Passes only devices specified by hostname or IP address.

- `CapabilityFilter`
  Passes only devices having the specified `capability` attribute within the NNM database; for example, `isRouter`.
- `OrganizationFilter`
  Passes only devices included in the specified customer model *organization* element.

The `InterfaceSelection` filter has one child element that further restricts the interfaces allowed to pass:

- `IPInterfaceFilter`
  Passes only interfaces specified by IP address.
- `OrganizationFilter`
  Passes only interfaces included in the specified customer model *organization* element.

When writing filters, use Perl5 regular expressions (see your Perl documentation for information about valid expressions). For example, *periods* in entries must be escaped with a backslash (\) character and the asterisks (*) must be escaped with a period (.):
`.*\.eagle\.wingnuts\.com`

When`IPHostFilter`, `CapabilityFilter`, `OrganizationFilter`, and/or `IPInterfaceFilter` are used, they contribute to the *intersection* of the `MgmtData` filter and the module filter. It is possible at runtime for the *intersection* of these lists to be the empty set. In this case, no nodes or interfaces are selected and no output occurs for this gauge.

When `IPHostFilter`, `CapabilityFilter`, `OrganizationFilter`, and/or `IPInterfaceFilter` are included yet left empty, they are considered an empty set which allows nothing to pass.

Leaving the `NodeSelection` filter or `InterfaceSelection` filter empty results in all nodes and interfaces that pass the `MgmtData` filter passing the current filter when determining the intersection of candidate nodes.

Establishing Global Settings for All Network Device Health Modules

This section focuses on the following two files:

- `/conf/share/modules/health/netHealthConfig.xml`
- `/conf/share/modules/NM/nmConfig.xml`

If you make changes to either one of these files, you must do the following:

WindowsNT/2000:
From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

UNIX:

As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

Stop on HP-UX: `/sbin/init.d/ovsip stop`
Start on HP-UX: `/sbin/init.d/ovsip start`

Stop on Solaris: `/etc/init.d/ovsip stop`
Start on Solaris: `/etc/init.d/ovsip start`

netHealthConfig.xml/dtd

This file contains configuration information that applies, or potentially applies, to all health gauges in all portal views. This information includes:

- Operational controls
- Health detail table settings
- Statistics definitions (Metric elements)
- Components and component groups that combine Metric elements
- Rating specifications for gathered statistics

See the `netHealthConfig.dtd` and `netHealthConfig.xml` files for more information.

Operational controls

The following attributes in the `netHealthConfig.xml` file control how the Network Device Health modules interact with NNM:

- `maxAge`
  SNMP data collected and supplied by NNM is checked to ensure that it is no older than the specified number of minutes. Older data is ignored for health calculation purposes. This defines "near real-time data." If young enough data does not exist for a particular metric, the metric is not used to compute overall health and the "Data Unavailable" message appears for that metric in Network Device Health drill-down tables.
- `rawDataRefresh`
  Sets the frequency with which raw data from NNM management stations should be updated. Expressed in minutes.

Health detail table settings

The `maxDetail` attribute sets the maximum number of devices allowed in the "detailed view" table at one time. If more devices pass the filtering requirements of a gauge, only the specified number of devices with the poorest health score rating are displayed. See also “Showing/Hiding the Health Details Tables” on page 82 and “Modifying the Health Details Tables” on page 84.

Statistics definitions (Metric elements)

There is one `Metric` element specifying each MIB object and each MIB expression used by any gauge. The `Metric` elements are reusable in multiple gauges. The `Metric` elements also define the rules for calculating the `score` associated with each returned MIB value. See also “Adding Statistics to the Health Calculation” on page 78.

Components and component groups that combine Metric elements

The `Component` elements control how the gauge's health score is calculated. The `Component` element assigns a weight to each `Metric`, as well as identifies whether or not this particular `Metric` is considered `vital` (if this `Metric` measures zero, the resource's health score is set to

zero regardless of other health score measures). See also “Modifying Weights Assigned to Values in Health Calculations” on page 79.

Rating specifications for gathered statistics

The `Rating` elements translate health *scores* to health *ratings*. The rating controls the color of the needle on the gauge and the color band around the outside edge of the gauge, as well as the health icons displayed in the health detail table. See also “Specifying Your Own Icons for the Details Table” on page 86.

nmConfig.xml/dtd

Only those NNM management stations or collection stations that are configured to do so in the `nmConfig.xml` file are allowed to provide Network Device Health information to SIP. See `nmConfig.xml` and `nmConfig.dtd` for more information. See also the SIP *Configuring NNM* manual (`Configuring_NNM.pdf`) file. The bold settings in the following example must be set before the Network Device Health Gauge module works:

```
<NNMStation
     hostname="hostname or IP address of the NNM system"
     snmpDataSource="get SNMP data collection data?: yes or no"
     alarmsDataSource="get NNM alarms?: yes or no"
     symbolRegSource="get OVW symbol registration information?: yes or no"
     webSrvPort="NNM web server port: usually 80 for NT; 8880 for Unix"
     ovwdbPort="ovwdb port: usually 9999 for NNM 6.1; 2447 for NNM 6.2"
     ovAlarmSrvPort="ovalarmsrv port: 2345 for NNM 6.1; 2953 for NNM 6.2"/>
```

The `snmpDataSource` attribute controls which NNM management stations are polled for current network health alarms.

When the NNM management station is running in a language other than English, no additional steps are required. See the "Internationalization" chapter of the SIP *Administrator Guide* (`Administrator_Guide.pdf`) is you want more information.

If SIP is set up to automatically configure NNM data collections, when multiple NNM management stations provide raw data to the Network Device Health module, duplication of SNMP data collections is avoided by sending collection requests for a given node to only one of the NNM management stations. The NNM management station chosen is that which first returned raw data for that node. See “Collecting Data for Network Device Health Gauges” on page 88 for more information.

Creating Your Own Network Device Health Gauge

You can create additional Network Device Health Gauges that display the combined results of the status information in NNM's database and the SNMP data collections of your choice from specified network devices. Please read the following topics before you start creating gauges:

- Overview for Creating Network Device Health Gauges
- Prerequisites to Creating Your Own Network Device Health Gauge
- Create Your Own Network Device Health Gauge

Overview for Creating Network Device Health Gauges

The components of Network Device Health Gauges are defined in the following files:

- `OVDefaultNetHealth.xml`

  This file defines the default set of health gauges for a new instance of the Network Device Health module. Each gauge is defined by a `Summary` element within this file. When a Network Device Health module is added to a tab with the `[Add]` button in the user interface, a *copy* of all gauges defined in the `OVDefaultNetHealth.xml` file is added to your current *`PortalView`*`.xml` file. These gauges show up in the selection list on the Network Device Health module's `Edit` page. You can show or hide any combination of the defined gauges.

TIP You can also customize specific instances of the gauges by directly editing *`PortalView`*`.xml` files.

- `netHealthConfig.xml`

  This file contains configuration information that applies, or potentially applies, to all health gauges for all portal users. It also contains a `Metric` element for each MIB object and each MIB expression included in any gauge `Summary` element. The `Metric` elements are reusable in multiple gauges. The `Metric` determines the

rules for calculating and displaying the `Score` associated with each returned MIB value. If you don't find a `Metric` element for the MIB object or MIB expression that you wish to use, you need to write one.

- `mibExprAuto.conf`

  This file defines SNMP MIB expressions used by the Service Information Portal. If you are using a MIB expression (mathematical formula comprised of MIB objects) that is not already defined in NNM, you need to define your MIB expression here.

- `snmpRepAuto.templ`

  This file is used by the `ovcolautoconf` program to automatically update NNM's Data Collector program to meet the current SIP requirements. Create an entry for each MIB object or MIB expression upon which you wish to collect data.

Prerequisites to Creating Your Own Network Device Health Gauge

Before creating your own network device health gauge:

1. Determine the set of nodes/interfaces from which this network device health gauge computes health scores. You will write filters to define your list. For information about available filters, see “Filtering Possibilities for the Network Device Health Module” on page 57. See "Filtering Data by Customer" in the SIP *Administrator Guide* (`Administrator_Guide.pdf`)for information about the `MgmtData` filter, customer model, and roles.
2. Determine which statistics should be used to compute health. Determine if the statistics can be provided by individual SNMP MIB objects or whether a mathematical formula using MIB objects is necessary.

   - **MIB Objects**

     MIB objects are attributes that an SNMP agent on a network device allows to be set or queried by an NNM management station. Currently, any MIB object that returns a numeric value is supported. (Strings are not supported.)

   - **MIB Expressions**

     MIB expressions are a feature of Network Node Manager that

allow for the creation of mathematical formulas comprised of MIB objects and explicit numeric values. MIB expressions allow you to derive more meaningful information than you could gather from individual MIB objects.

NNM provides a variety of predefined MIB expressions. In addition, the Service Information Portal provides the following MIB expressions for interfaces: Interface % Utilization (`p_if%util`), Interface % Inbound Errors (`p_if%inerrors`), Interface % Outbound Errors (`p_if%outerrors`). MIB expressions provided with the Service Information Portal for nodes are: Cisco CPU Utilization (`p_if%p_cisco5minavgbusy`).

Service Information Portal preconfigured MIB expressions are defined in the following file on the NNM management station:

— *Windows NT/2000:*

`<NNM_install_dir>\conf\ovcolautoconf\mibExprAuto.conf`

— *UNIX:*

`/etc/opt/OV/share/conf/ovcolautoconf/mibExprAuto.conf`

See “Manually Configuring NNM’s Data Collector to Provide the Required SIP Data” in the SIP *Configuring NNM* manual (`Configuring_NNM.pdf`) to learn about each MIB expression provided. Information about writing your own MIB expression is presented in the following pages.

Create Your Own Network Device Health Gauge

To create your own network device health gauge, complete the following steps. Note that some steps are carried out on the SIP server and others on each NNM management station that supplies data for the new gauge.

Before you start this series of steps, complete the prerequisite steps and have your list of nodes and interfaces to be monitored and your list of statistics to be gathered.

Steps on each NNM management station

1. Check NNM's Data Collection configuration to see if the statistics that you need are already being collected. If you find all the statistics that you need, skip to "Steps on the SIP server" on page 66:

   From any NNM submap, select `Options:Data Collections & Thresholds`. Review the list of currently configured collections. See *Managing Your Network with NNM* for more information. See also the Help information from within the Data Collections & Thresholds window.

2. Make sure that the MIB specification files, whose objects you wish to use, are loaded into NNM. From any NNM submap, select `Options:Load/Unload MIBs`. See *Managing Your Network with NNM* for more information about loading MIB specification files into NNM.

3. OPTIONAL: Write your own MIB expression and add it to the `mibExprAuto.conf` file (or copy and modify one of the MIB expressions supplied). For information about writing MIB expressions, please see *Managing Your Network with NNM*, and the *mibExpr.conf* and the *mib.coerce* reference pages in NNM's online help (or the UNIX manpages).

   Before adding your new MIB expression to the `mibExprAuto.conf` file, save a copy of the original file. After writing your new MIB expression, you must load the new expression into NNM by typing the following at the command prompt on the NNM management station. This command checks the syntax of your MIB expression and forces an update to NNM's `mibExpr.conf` file which allows data collections to be enabled:

   - *Windows NT/2000:*

```
xnmcollect -loadExpr <NNM_install_dir>\conf\ovautocolconf\mibExprAuto.conf
```

   - *UNIX:*

```
xnmcollect -loadExpr /etc/opt/OV/share/conf/ovautocolconf/mibExprAuto.conf
```

4. Modify the `snmpRepAuto.templ` file to enable automatic configuration of NNM Data Collector. This ensures that the SNMP data needed to drive your gauges is available. For information about the attributes needed for each data collection, see "snmpRepAuto.templ" on page 92.

When executed, the `ovcolautoconf.exe` program uses this file to configure the `snmpRep.conf` file with the most recent SIP data collection requirements. This keeps NNM's Data Collector in sync with changes in the *`PortalView`*`.xml` files. For more information about how the data collection process works, see "Collecting Data for Network Device Health Gauges" on page 88.

Steps on the SIP server

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

1. Check the `netHealthConfig.xml` file.

   Make sure that there is a `Metric` element for each MIB object and each MIB expression that you intend to use. If not, add a `Metric` element to the `netHealthConfig.xml` file that defines the rules for determining the `score` associated with each newly defined MIB

value. See the `netHealthConfig.dtd` file and `netHealthConfig.xml` file for information about the attributes needed for each `Metric` element. Example:

```
<Metric id="IfUtil" title="Interface Utilization" autoConfig="yes"
   href="snmp://%item%[IfIndex]/V_If%util">
      <Scale lower="0"  upper="25" translation="100"/>
      <Scale lower="25" upper="50" translation="70"/>
      <Scale lower="50" upper="75" translation="40"/>
      <Scale lower="75" upper="100" translation="0"/>
</Metric>
```

2. After modifying the `netHealthConfig.xml` file, do the following:

   WindowsNT/2000:
   From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

   UNIX:

   As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **`/sbin/init.d/ovsip stop`**
   Start on HP-UX: **`/sbin/init.d/ovsip start`**

   Stop on Solaris: **`/etc/init.d/ovsip stop`**
   Start on Solaris: **`/etc/init.d/ovsip start`**

3. Modify the `OVDefaultNetHealth.xml` file by adding a `Summary` element that defines the new network device health gauge. See the `OVNetworkHealth.dtd` file for information about the attributes needed within each `Summary` element. Example:

```
<Summary title="Access Link Health" display="yes" displayDepth="3">
   <Component weight="2" href="#IfStatus"/>
   <Component weight="1" href="#IfUtil"/>
   <Component weight="1" href="#IfInErrors"/>
   <Component weight="1" href="#IfOutErrors"/>
   <InterfaceSelection title="Access Links" id="AccessLinks" op="AND">
        Your filters would be defined here
   </InterfaceSelection>
</Summary>
```

4. You are now ready to insert the new gauge into any *PortalView*.xml file.

   - Existing Network Device Health module instances in *PortalView*.xml files: copy your new gauge's Summary element from the OVDefaultNetHealth.xml file, then open the *PortalView*.xml file and paste the new gauge's Summary into the desired location.
   - New Network Device Health module instances: after creating a new *PortalView*.xml file, log into the portal view and navigate to the desired tab. Select Network Device Health in the list of available modules and click [Add]. If necessary, click the [Edit] button in the title bar of the newly added Network Device Health module, and select the new gauge from the list of available gauges.

NOTE

You can edit the *PortalView*.xml file directly after inserting the new gauge and modify the Summary element to further customize the desired results. See "Filtering Possibilities for the Network Device Health Module" on page 57.

Editing Network Device Health Gauge Modules

- Adding a Network Device Health Module to a Portal View
- Displaying Additional Health Gauges in a Network Device Health Module
- Changing the Display Order of Health Gauges
- Removing Health Gauges from a Network Device Health Module
- Adding Statistics to the Health Calculation
- Modifying Weights Assigned to Values in Health Calculations
- Controlling How Health Is Calculated
- Showing/Hiding the Health Details Tables
- Modifying the Health Details Tables

The following Network Device Health gauges are preconfigured in the template files provided with HP OpenView Service Information Portal. You can insert the code for these gauges into a portal view and display your choice of gauges. If desired, edit each *PortalView*.xml file so that only the devices of your choosing are included in the gauge's calculations:

- **Router Health**
  This gauge monitors every device in the NNM database that has the `isRouter` capability. If you want to narrow the set of included routers, see "Filtering Possibilities for the Network Device Health Module" on page 57. If you want to expand the set of included routers, open NNM and configure the `isRouter` capability for the additional devices.
- **Server Health**
  This gauge monitors every device in the NNM database that has the `isServer` capability. If you want to narrow the set of included servers, see "NodeSelection and InterfaceSelection Filters for Specific Health Gauges" on page 57. If you want to expand the set of included servers, open NNM and configure the `isServer` capability for the additional devices.

- **Key Device Health**
  This gauge monitors every device in the NNM database that has the `isKeyDevice` capability. HP OpenView Customer Views introduces the `isKeyDevice` capability; you can set the `isKeyDevice` attribute for any device in NNM. Within Customer Views, see the online help or the web-based *Concepts Guide* for more information. For more information, see “Filtering Possibilities for the Network Device Health Module” on page 57.
- **CPE Health**
  This gauge monitors every device in the NNM database that has the `isCPE` (customer premises equipment) capability. HP OpenView Customer Views introduces the `isCPE` capability, you can set the `isCPE` attribute for any device in NNM. Within Customer Views, see the online help or the web-based *Concepts Guide* for more information. For more information, see “Filtering Possibilities for the Network Device Health Module” on page 57.
- **Interface Health**
  This gauge cannot operate unless a specific list of interfaces is established through at least one of the filtering levels allowed within Service Information Portal configuration. This limitation is imposed so that you don't accidently set up data collections on every interface in the whole management domain. See “Filtering Possibilities for the Network Device Health Module” on page 57 and see"Filtering Data by Customer" in the SIP *Administrator Guide* (`Administrator_Guide.pdf`)for information about the `MgmtData` filter, customer model, and roles.

NOTE Server, Key Devices, CPE, and Interface gauges display the error message “Currently not Configured” or “Managed Objects Not Found” until you perform the configurations described above.

You can also write your own Network Device Health gauges to monitor and calculate whatever you want. See “Creating Your Own Network Device Health Gauge” on page 62.

Adding a Network Device Health Module to a Portal View

Two ways exist for adding Network Device Health modules to a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *PortalView*.xml file you want to customize.
2. Navigate to the appropriate Tab.
3. At the bottom of the window, select Network Device Health and click [Add].
4. Save the changes and return to the main portal page; click the [OK] button.

TIP If you want to create and add a different instance of a module to the list of available modules, see the SIP *Administrator Guide* (Administrator_Guide.pdf)

Through the Portal View File

Modules can be added and configured by directly editing a PortalView.xml file. Modules are wrapped in the ModuleInstance element. The ModuleInstance id must be unique among all module instances in the portal view file. For information about the ModuleInstance element, see the SIP *Administrator Guide* (Administrator_Guide.pdf), "Designing Portal Views."

Follow the directions in the SIP *Administrator Guide*.

Refer to the OVNetworkHealth.dtd file for more information.

You can copy and paste the contents of the following file into your *PortalView*.xml file as a starting point:

- OVDefaultNetHealth.xml

Displaying Additional Health Gauges in a Network Device Health Module

Two ways exist for displaying health gauges in a portal view:

- Through the User Interface
- Through the Portal View File

Gauges that are displayed for the first time are not fully functional until a collection cycle runs on NNM. See "Collecting Data for Network Device Health Gauges" on page 88.

Through the User Interface

1. Log into the *PortalView*.xml file you want to customize.
2. On the title bar of the Network Device Health module, click [Edit].
3. On the Network Device Health - Edit page, select Choose from List.
4. Select a health category from the Available Health Categories list and click the [Add] button.
5. Repeat step 4 until the Displayed Health Categories list contains all the categories you want to display.
6. Save the changes and return to the main portal page; click the [OK] button.

Your changes are not fully functioning until the next data collection configuration update occurs.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the file that contains the Network Device Health module you wish to adjust:
   - *Windows NT/2000:*

<*SIP_install_dir*>\registration\defaults\OVDefaultNetHealth.xml or
<*SIP_install_dir*>\conf\share\views*PortalView*.xml

- *UNIX:*

`/etc/opt/OV/SIP/registration/defalts/OVDefaultNetHealth.xml or`
`/etc/opt/OV/SIP/conf/share/views/`*`PortalView`*`.xml`

2. Locate the `Summary` element of the gauge you wish to display or hide, for example:

```
<Summary display="no" displayDepth="3" id="CPEHealth" title="CPE Health">
```

3. Set the display attribute:

   `display="yes"` to display the gauge

   `display="no"` to hide the gauge

4. Save the XML file. In a browser, log in as the appropriate customer and ensure that the desired behavior has been established.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Changing the Display Order of Health Gauges

Two ways exist for changing the display order of health gauges:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *PortalView*.xml file you want to customize.
2. On the title bar of the Network Device Health module, click [Edit].
3. On the Network Device Health - Edit page, select Choose from List.
4. Rearrange the order of the Displayed Health Categories by selecting a health category and clicking the [Up] or [Down] button.
5. Repeat step 4 until the Displayed Health Categories list displays the health categories in the order you prefer.
6. Save the changes and return to the main portal page; click the [OK] button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the file that contains the Network Device Health module you wish to adjust:
   - *Windows NT/2000:*

<*SIP_install_dir*>\registration\defaults\OVDefaultNetHealth.xml or
<*SIP_install_dir*>\conf\share\views*PortalView*.xml

   - *UNIX:*

/etc/opt/OV/SIP/registration/defalts/OVDefaultNetHealth.xml or
/etc/opt/OV/SIP/conf/share/views/*PortalView*.xml

2. For the gauge you wish to move, cut everything from `<Summary title=` through `</Summary>` and paste it into the desired location between the `<ModuleInstance id="NetworkHealth"` and the `</ModuleInstance>` tags.
3. Repeat step 2 until the *`PortalView`*`.xml` file contains the health gauge summaries in the order in which you want to display them to this customer.
4. Save the XML file. In a browser, log in as the appropriate customer and ensure that the desired behavior has been established.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Removing Health Gauges from a Network Device Health Module

Two ways exist for removing Network Device Health gauges from a portal view:

- Through the User Interface
- Through the Portal View File

Data collection for a specific gauge is discontinued once the gauge has not been displayed in any portal view for 30 days. See "Collecting Data for Network Device Health Gauges" on page 88.

Through the User Interface

1. Log into the *PortalView*.xml file you want to customize.
2. On the title bar of the Network Device Health module, click [Edit].
3. On the Network Device Health - Edit page, select Choose from List.
4. Select a health category in the Displayed Health Categories list, and click the [Remove] button.

   (This changes the display attribute in the gauge's Summary element in the *PortalView*.xml file so that the gauge is no longer visible to your customer. The Summary element is still present in the *PortalView*.xml file.)
5. Repeat step 4 until the Displayed Health Categories list contains only the health categories you want to display.
6. Save the changes and return to the main portal page; click the [OK] button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the file that contains the Network Device Health module you wish to adjust:

- *Windows NT/2000:*

`<SIP_install_dir>\registration\defaults\OVDefaultNetHealth.xml or`
`<SIP_install_dir>\conf\share\views\PortalView.xml`

- *UNIX:*

`/etc/opt/OV/SIP/registration/defalts/OVDefaultNetHealth.xml or`
`/etc/opt/OV/SIP/conf/share/views/PortalView.xml`

2. Search the string `<Summary title=` until you locate the network device health gauge that you wish to remove.
3. Do one of the following:
   - Change the `display` attribute in the gauge's `Summary` element to `display="no"`. The `Summary` element is still present in the XML file.
   - Delete everything from `<Summary title=` through `</Summary>` to remove the selected gauge from the customer's configuration file.
4. Repeat steps 2-3 until either only the health gauge summaries you want to display to this customer are set to `display="yes"` or only the health gauge summaries you want to display to this customer remain in this XML file.
5. Save the XML file. In a browser, log in as the appropriate customer and ensure that the desired behavior has been established.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer

which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Adding Statistics to the Health Calculation

If you wish to add statistical items to a gauge's calculation of network device health:

1. Determine which statistics should be added to the health gauge's calculations. Determine if each statistic can be provided by an individual SNMP MIB object or whether a mathematical formula using MIB objects is necessary.

   - **MIB Objects**

     MIB objects are attributes that an SNMP agent on a network device allows to be set or queried by an NNM management station. Currently, any MIB object that returns a numeric value is supported. (Strings are not supported.)

   - **MIB Expressions**

     MIB expressions are a feature of Network Node Manager that allow for the creation of mathematical formulas comprised of MIB objects. MIB expressions allow you to derive more meaningful information than you could gather from individual MIB objects.

     Please see *Managing Your Network with NNM*. See also the *mibExpr.conf* and the *mib.coerce* reference pages in NNM's online help (or the UNIX manpages) for information about writing MIB expressions.

2. Follow the directions in "Creating Your Own Network Device Health Gauge" on page 62.

Modifying Weights Assigned to Values in Health Calculations

The `weight` attribute allows you to control how much emphasis is placed upon each `Component` (MIB object, MIB expression, or device status) within a gauge. For example, if an interface's status is *down*, it has more of an impact on the device health calculation than a high utilization measurement.

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the file that contains the Network Device Health module you wish to adjust:

   - *Windows NT/2000:*

   `<SIP_install_dir>\registration\defaults\OVDefaultNetHealth.xml or`
   `<SIP_install_dir>\conf\share\views\PortalView.xml`

   - *UNIX:*

   `/etc/opt/OV/SIP/registration/defalts/OVDefaultNetHealth.xml or`
   `/etc/opt/OV/SIP/conf/share/views/PortalView.xml`

2. Locate the `Summary` element for the network device health gauge whose weights you wish to change.

   Refer to the `OVNetworkHealth.dtd` file for more information.

3. Locate the gauge's `Summary` element and change the `weight` attributes. Example:

```
<Summary display="no" displayDepth="3" id="InterfaceHealth" title="Interface
Health">
     <Component href="#IfStatus" vital="yes" weight="2"/>
     <Component href="#IfUtil" vital="no" weight="1"/>
     <Component href="#IfInErrors" vital="no" weight="1"/>
     <Component href="#IfOutErrors" vital="no" weight="1"/>
     <InterfaceSelection id="AllInterfaces" op="AND" title="All Interfaces"/>
</Summary>
```

4. Save the XML file. In a browser, log in as the appropriate customer and ensure that the desired behavior has been established.

NOTE After you make modifications to XML files, validate the syntax. Provided

with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Controlling How Health Is Calculated

There are five steps in determining health. You can make modifications to any combination of these steps.

1. Data is gathered through HP OpenView Network Node Manager and the requested *values* are returned to the Service Information Portal.

   For more information about controlling this process, see "Collecting Data for Network Device Health Gauges" on page 88.

2. For SNMP data, the returned values are checked to ensure that they are valid by noting how many minutes have passed since they were collected by NNM. This is controlled by the `maxAge` attribute in the `netHealthConfig.xml` file. Example:

```
<NetworkHealthConfig maxDetail="20" maxAge="60">
```

   The `maxAge` attribute affects all health gauges defined within all portal views.

3. The `Metric` element (in the `netHealthConfig.xml` file) for the requested MIB object or MIB expression converts the returned *value* to a `Score` (0-100).

4. Each gauge's `Summary` element assigns a `weight` to each `Score`. Based upon the weighted average (mean) of these scores, a health score is computed for each node or interface represented by the gauge.

5. A gauge's health score is computed as the average (mean) score of all nodes/interfaces represented by the gauge. Score is converted to a *rating* as defined in the `Rating` element of the `netHealthConfig.xml` file. Example:

```
<Rating>
   <Scale lower="1" upper="1"    tranlation="unknown"/>
   <Scale lower="0" upper="40"   tranlation="critical"/>
   <Scale lower="40" upper="80"  tranlation="minor"/>
   <Scale lower="80" upper="100" tranlation="normal"/>
</Rating>
```

   The `Rating Scale` controls where the needle of the gauge is placed.

   The rating scale also controls the color bands on the gauges themselves and, in the detail tables, controls which icon is associated with different ratings. The rating scale affects all gauges defined within any portal.

6. If you make changes to any of the files listed in these steps, you must do the following before SIP acknowledges the changes:

   WindowsNT/2000:
   From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

   UNIX:

   As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **`/sbin/init.d/ovsip stop`**
   Start on HP-UX: **`/sbin/init.d/ovsip start`**

   Stop on Solaris: **`/etc/init.d/ovsip stop`**
   Start on Solaris: **`/etc/init.d/ovsip start`**

NOTE After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML

parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Showing/Hiding the Health Details Tables

You can control whether or not drill-down access is granted to the Details Tables of a specific Network Device Health gauge by setting the `displayDepth` attribute.

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the file that contains the Network Device Health module you wish to adjust:

    - *Windows NT/2000:*

`<SIP_install_dir>\registration\defaults\OVDefaultNetHealth.xml or`
`<SIP_install_dir>\conf\share\views\PortalView.xml`

    - *UNIX:*

`/etc/opt/OV/SIP/registration/defalts/OVDefaultNetHealth.xml or`
`/etc/opt/OV/SIP/conf/share/views/PortalView.xml`

2. Locate the gauge's `Summary` element (search on the title). For example:

```
<Summary title="Router Health" display="yes" displayDepth="3">
```

3. Locate the `displayDepth` attribute.
4. Designate the appropriate setting:

   1=gauge only (no drill-down)

   2=drill down to node-only or interface-only details

   3=drill down to node *and* drill down to interface details (if available)
5. Save the XML file. In a browser, log in as the appropriate customer and ensure that the desired behavior has been established.

   See also "Health detail table settings" on page 60 for information about the `maxDetail` attribute in the `netHealthConfig.xml` file.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Modifying the Health Details Tables

You can control the information displayed in the details tables in the following ways:

- Global Health Detail Table Settings
- Specifying Your Own Icons for the Details Table
- Per-Module-Instance Health Detail Table Settings

Global Health Detail Table Settings

You can globally control `maxAge`, `maxDetail`, `Ratings`, and icons used in any Details Tables of all Network Device Health gauges defined in any XML file.

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. With an ASCII or XML editing program, open the `netHealthConfig.xml` file:

   - *Windows NT/2000:*

`<SIP_install_dir>\conf\share\modules\health\netHealthConfig.xml`

   - *UNIX:*

`etc/opt/OV/SIP/conf/share/modules/health/netHealthConfig.xml`

2. Locate the `NetworkHealthConfig` element. For example:

   ```
   <NetworkHealthConfig maxDetail="20" maxAge="60">
   ```

3. Designate the appropriate settings:

   `maxDetail` designates the maximum number of rows allowed in the detailed view table at one time. If more devices pass the filtering requirements of a gauge, only the specified number of devices with the poorest health score rating are displayed.

   `maxAge` controls the number of *minutes* that can pass since the last NNM data collection cycle before the data is considered obsolete. Older data is ignored for health calculation purposes. This defines *near real-time data*.

4. Designate the appropriate rating scale. The ratings control the color of the needle on the gauge and color changes of the band around the

outside edge of the gauge, as well as the health icons displayed in the health detail table.

The `Rating Scale` is used to translate the *scores* to health *ratings*. For example:

```
<Rating>
   <Scale lower="1" upper="1"    tranlation="unknown"/>
   <Scale lower="0" upper="40"   tranlation="critical"/>
   <Scale lower="40" upper="80"  tranlation="minor"/>
   <Scale lower="80" upper="100" tranlation="normal"/>
</Rating>
```

5. *WindowsNT/2000*:
   From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

   UNIX:

   As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **`/sbin/init.d/ovsip stop`**
   Start on HP-UX: **`/sbin/init.d/ovsip start`**

   Stop on Solaris: **`/etc/init.d/ovsip stop`**
   Start on Solaris: **`/etc/init.d/ovsip start`**

6. Save the XML file. In a browser, log in as any customer and ensure that the desired behavior has been established.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`

UNIX: /opt/OV/SIP/bin

If the output of the xmlvalidate command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at www.xmlspy.com.

Specifying Your Own Icons for the Details Table

To change the icons in the Network Device Health detail tables, change the images in the following directory. Maintain the names of the images:
Windows NT/2000: <*SIP_install_dir*>\htdocs\C\images\health\
UNIX: /opt/OV/SIP/htdocs/C/images/health/

Per-Module-Instance Health Detail Table Settings

You can control the showRawData and showUnknown attributes in the Details Tables of all gauges within a specific Network Device Health module instance.

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the file that contains the Network Device Health module you wish to adjust:

   - *Windows NT/2000:*

<*SIP_install_dir*>\registration\defaults\OVDefaultNetHealth.xml or
<*SIP_install_dir*>\conf\share\views*PortalView*.xml

   - *UNIX:*

/etc/opt/OV/SIP/registration/defalts/OVDefaultNetHealth.xml or
/etc/opt/OV/SIP/conf/share/views/*PortalView*.xml

2. Locate the NetworkHealth element. For example:

   ```
   <NetworkHealth showRawData="no" showUnknown="no">
   ```

3. Designate the appropriate settings (any of the following are valid entries: yes | no | YES | NO | 1 | 0):

   showRawData toggles on/off the display of the columns displaying data

derived from each `Metric` element referenced in each child `Component` element of every gauge `Summary` element included in the current module instance. The default setting is to hide these columns and only display the final health score.

`showUnknown` controls whether or not nodes or interfaces whose health score cannot be computed (usually, because that object's status in NNM's object database is set to `unknown`) are included in the gauge's details tables. The default is to exclude information derived from devices with `"unknown"` status. If this attribute is `"on"`, rows for nodes/interfaces with `"unknown"` health status are added to the end of the detail table if the `maxDetail` attribute allows enough room.

4. Save the XML file. In a browser, log in as the appropriate customer and ensure that the desired behavior has been established.

TIP If you change these settings in the `OVDefaultNetHealth.xml` file, any *`PortalView`*`.xml` files that you create in the future will have the settings that you designated as their default setting.

NOTE After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Collecting Data for Network Device Health Gauges

HP OpenView Network Node Manager (NNM) collects all SNMP data requested by HP OpenView Service Information Portal (SIP) and provides current information about device status.

Network Device Health gauges calculate the health of specific network devices using information gathered by NNM management stations. Changes are visible in the SIP's Network Device Health gauges each time the portal view is displayed or refreshed.

SIP depends upon two programs that reside on each NNM management station (`getnnmdata.exe` and `ovcolautoconf.exe`) to collect requested data:

1. Each time a Network Device Health gauge is displayed, SIP logs the underlying data requests. A list of requested MIB objects and MIB expressions from any Network Device Health module gauge is compiled by SIP. The list documents which MIB objects and MIB expressions are being requested for which network devices from which NNM management stations.

NOTE

The underlying MIB objects and MIB expressions appear in Network Device Health gauge definitions as the `Component` elements' `href` attributes. Each `href` attribute must have a corresponding `Metric` element defined in the `netHealthConfig.xml` file that specifies exactly which MIB object or MIB expression is being requested.

2. SIP contacts the `getnnmdata.exe` on each NNM management station that is configured in the `nmConfig.xml` file. The frequency of this action is determined by the `rawDataRefresh` parameter setting in the `netHealthConfig.xml` file on the SIP server (by default, every 10 minutes).
3. SIP receives the most recent data collection results from the NNM database. SIP also places the current request log file in the `ovcolautoconf` directory. Requests from each SIP server are gathered here (`dc.needs<SIPserverIPaddress>`).

TIP You must create the `ovcolautoconf` directory before this step works. See *Configuring NNM* (`Configuring_NNM.pdf`) for more information.

4. To complete the automatic configuration process, run the `ovcolautoconf.exe` command. The `ovcolautoconf` command must be executed on the NNM management station, either manually or as a scheduled task that you define. `ovcolautoconf` does the following:
   - All SIP servers' data collection needs are processed. The list of data collection requests is configured using the information in `snmpRepAuto.templ` file and placed in the `snmpRepPrev.conf` file.
   - If necessary, NNM's Data Collector configurations are updated by making SIP additions or changes to the `snmpRep.conf` file.
   - Data collections are configured on an *as-needed* basis, rather than a *potentially* needed basis. In other words, until a gauge is displayed in a portal view, no data collection is initiated. If a gauge is not displayed for 30 days (default setting), the data collections are discontinued.
5. The `snmpRep.conf` file is used by the SNMP Data Collector as a guide for gathering data. The entries from the HP OpenView Service Information Portal do not interfere with data collection configurations that were entered directly through NNM. The `ovcolautoconf.exe` deletes any data collection configurations that are no longer needed (provided they are not needed by other OpenView products). The collected data can be automatically trimmed from NNM's databases after it ages for one week (depending upon the settings in NNM's reporting feature).

TIP See the SIP *Configuring NNM* manual (`Configuring_NNM.pdf`) for important additional information about the SIP data collection process for the Network Device Health module.

mibExprAuto.conf

This file resides on the NNM management station. It contains the MIB expression definitions that are being used by SIP for Network Device Health calculations. MIB expressions are a feature of Network Node Manager that allow for the creation of mathematical formulas comprised of MIB objects. MIB expressions allow you to derive more meaningful information than you could gather from individual MIB objects.

Service Information Portal preconfigured MIB expressions are defined in this file (see "Manually Configuring NNM's Data Collector to Provide the Required SIP Data" section in the *Configuring NNM*, `Configuring_NNM.pdf`, for information about installing this file onto your NNM management station). More information is available within the following file, itself:

- *Windows NT/2000:*

`<NNM_install_dir>\conf\ovcolautoconf\mibExprAuto.conf`

- *UNIX:*

`/etc/opt/OV/share/conf/ovcolautoconf/mibExprAuto.conf`

OPTIONAL: If you need to write your own MIB expression:

1. Before you modify the `mibExprAuto.conf` file, save a copy of the original.
2. Add your MIB expression to the `mibExprAuto.conf` file (TIP: copy and modify one of the MIB expressions supplied).

   For information about writing MIB expressions, see *Managing Your Network with NNM*. See also the *mibExpr.conf* and the *mib.coerce* reference pages in NNM's online help (or the UNIX manpages).
3. Verify that the MIB files, whose objects you wish to use, are loaded into NNM. See *Managing Your Network with NNM* for more information about loading MIB files into NNM.

4. After writing your new MIB expression, you must load the new expression into NNM by typing the following at the command prompt. This command checks the syntax of your MIB expression and forces an update to NNM's `mibExpr.conf` file which allows data collections to be enabled:

   - *Windows NT/2000:*

```
xnmcollect -loadExpr <NNM_install_dir>\conf\ovcolautoconf\mibExprAuto.conf
```

   - *UNIX:*

```
xnmcollect -loadExpr /etc/opt/OV/share/conf/ovcolautoconf/mibExprAuto.conf
```

5. Make a new entry into the `snmpRepAuto.templ` file so that NNM could begin collecting the requested information (see "snmpRepAuto.templ" on page 92).

6. Review the following section for possible additional required steps: "Create Your Own Network Device Health Gauge" on page 64.

snmpRepAuto.templ

This file exists on each NNM management station. (See *Configuring NNM* (Configuring_NNM.pdf) for installation instructions.)If you create any new gauges, you must ensure that there is one entry in the snmpRepAuto.templ file for each MIB object and each MIB expression that needs to be collected. (See "Location of Relevant Files" on page 95.)

To view the list of configured collections and make any necessary additions, at the command line type the following:

- *Windows NT/2000:*

  ```
  xnmcollect -snmpColConfFile snmpRepAuto.templ
  ```

- *UNIX:* log in as root and then type,

  ```
  xnmcollect -snmpColConfFile snmpRepAuto.templ
  ```

Review the list. In the Source field you will see the variable _NODE_, which is automatically replaced with any specific devices requested by SIP.

If you do not see each MIB object and/or MIB expression that you are using in your gauge, create a new Data Collector entry:

1. Highlight any MIB Object in the top half of the window and select Edit:MIB Object->Copy.
2. Select the new MIB object or MIB expression that you wish to collect data upon.
3. You can change the collection interval setting, otherwise leave the settings as they are. You should see the variable _NODE_ in the Source field.

ovcolautoconf.exe

ovcolautoconf.exe configures the NNM SNMP Data Collector (snmpCollect) to gather data requested by the HP OpenView Service Information Portal (SIP).

SYNOPSIS

ovcolautoconf [-verbose] [-outfile <filename>] [-maxConfAge <#ofdays>]

DESCRIPTION

ovcolautoconf is a Network Node Manager (NNM) command that configures the NNM SNMP Data Collector (snmpCollect) to gather data requested by SIP. If invoked without the -outfile option, ovcolautoconf updates NNM's data collection configuration to reflect SIP SNMP data needs. Specifically, ovcolautoconf processes SIP server configuration request files found in $OV_DB/snmpCollect/ovcolautoconf. These files have names of the form dcNeeds.<SIP Server IP Addr>. {For information about how these request files are placed in this directory, see the Service Information Portal manual "*Configuring NNM*"). The template file $OV_CONF/ovcolautoconf/snmpRepAuto.templ is used to construct data collector configuration entries corresponding to these requests. The configuration entries are then loaded into the data collector configuration file $OV_CONF/snmpRep.conf, and snmpCollect is notified that its configuration has been modified. ovcolautoconf truncates the SIP server request files after successfully processing them. The most recent data collector configuration submitted by ovcolautoconf can be found in the file $OV_DB/snmpCollect/ovcolautoconf/snmpRepPrev.conf

If the data collection configuration needs have not changed since the last execution of ovcolautoconf, no changes to snmpRep.conf are made and no reconfiguration event is sent to snmpCollect.

ovcolautoconf automatically removes data collector configuration entries are no longer needed by SIP. See the discussion of the -maxConfAge option below for details

OPTIONS

`-maxConfAge <#ofdays>` Removes configuration entries that have gone unrequested for the specified number of days. Applies only to configuration entries submitted by `ovcolautoconf`. Default is 30 days.

`-outfile <filename>` Don't update NNM's data collection configuration, but instead write the configuration to the specified file.

`- verbose` Send verbose output, including notification of configuration entries that have been aged out, to `stdout`.

TROUBLESHOOTING

Warning and error messages are sent to `stderr`.

FILES ON THE NNM MANAGEMENT STATION

Windows NT/2000:

`<NNM_install_dir>\conf\ovcolautoconf\snmpRepAuto.templ`

`<NNM_install_dir>\conf\ovcolautoconf\mibExprAuto.conf`

`<NNM_install_dir>\databases\snmpCollect\ovcolautoconf\snmpRepPrev.conf`

`<NNM_install_dir>\databases\snmpCollect\ovcolautoconf\dcNeeds.<SIPserverIPaddress>`

UNIX:

`/etc/opt/OV/share/conf/ovcolautoconf/snmpRepAuto.templ`

`/etc/opt/OV/share/conf/ovcolautoconf/mibExprAuto.conf`

`/var/opt/OV/share/databases/snmpCollect/ovcolautoconf/snmpRepPrev.conf`

`/var/opt/OV/share/databases/snmpCollect/ovcolautoconf/dcNeeds.<SIPserverIPaddress>`

See also the *ovrequestd*, *snmpCollect*, *snmpCol.conf*, *mibExpr.conf*, and the *mib.coerce* reference pages in NNM's online help (or the UNIX manpages) for information about NNM's data collection process.

Location of Relevant Files

Table 3-4 Network Device Health Module Files on the SIP Server

File Name	Windows NT/2000 Location <*SIP_install_dir*>/SIP/....	UNIX Location etc/opt/OV/SIP/....
nmConfig.dtd	conf/share/modules/NM/	conf/share/modules/NM/
nmConfig.xml	conf/share/modules/NM/	conf/share/modules/NM/
OVModuleRegistration.dtd	registration/	registration/
OVRegNetHealth.xml	registration/	registration/
netHealthConfig.dtd	conf/share/modules/health/	conf/share/modules/health/
netHealthConfig.xml	conf/share/modules/health/	conf/share/modules/health/
OVNetworkHealth.dtd	conf/share/views/	conf/share/views/
OVDefaultNetHealth.xml	registration/defaults/	registration/defaults/
PortalView.xml	conf/share/views	conf/share/views

Table 3-5 Data Collection Process Files on the NNM Management Station

File Name	Windows NT/2000 Location <*NNM_install_dir*>/ ...	UNIX Location....
getnnmdata.exe	www/cgi-bin/	/opt/OV/www/cgi-bin/
dcNeeds.<SIPserver>	databases/ snmpCollect/ovcolautoconf/	var/opt/OV/share/databases/ /snmpCollect/ovcolautoconf/
ovcolautoconf.exe	bin/	opt/OV/bin
snmpRepAuto.templ	conf/ovcolautoconf/	etc/opt/OV/share/conf/ ovcolautoconf
mibExprAuto.conf	conf/ovcolautoconf/	etc/opt/OV/share/conf/ ovcolautoconf

4 Topology Module

Understanding Topology Data

The Topology module displays one or more Network Node Manager (NNM) submaps. Submaps provide a graphical view of the network environment or system management information. Each submap displays a different perspective of the environment. You may be able to display another submap by clicking on a symbol; for example display a submap showing all interfaces within a router by clicking on the router symbol. Click the browser's [Back] button to return to the previous submap.

Each submap that you display is associated with an NNM management station and a map. NNM management stations provide and maintain the operational SNMP/network management information.

Several things are important to know about the submaps displayed through the Topology module:

- NNM must be configured for use with the Topology module. For detailed information, see *Configuring NNM*. An electronic version of the guide is located in the following directory:

  Windows NT/2000: <*SIP_install_dir*>\htdocs\C\manuals\NNM\Configuring_NNM.pdf

  UNIX:
  /opt/OV/SIP/htdocs/C/manuals/NNM/Configuring_NNM.pdf

- Submaps that are targeted within Topology modules must either be currently displayed on the NNM management station or be configured as *persistent* (not *transient*) within NNM before they display in the portal. This means that the submaps must be stored in RAM on the NNM management station and not generated on-the-fly upon request.
- Submaps that are accessed through drill-down (optional behavior, default = no drill-down) might be *transient* within NNM, depending upon the global settings you choose.
- If your submaps have *auto-layout* turned off in NNM, the New Object Holding Area does not display in SIP. You must move symbols out of the New Object Holding Area to make them visible in SIP.
- The submaps displayed in SIP are actually completely new redrawn versions. Outer shapes for symbols are not dynamically generated. SIP supported outer shapes are circle, square, diamond, hexagon and

octagon. Square is the generic shape for any symbol from NNM that uses a shape that is unsupported in SIP.

- If you are using HP OpenView Customer Views software on the NNM management station, you might have multiple submaps with the same name. Be careful to specify which instance of that submap name you wish to access in SIP. If you use the SIP user interface to select submaps, SIP determines which submaps have duplicate names and generates a selection list with paths included.

You control how NNM submaps are displayed by configuring the following files:

- `/registration/OVRegTopology.xml`
  Registers the Topology module so that SIP has access. Specifies various properties required by SIP. (See the `OVModuleRegistration.dtd` file for more information.)
- `/registration/defaults/OVDefaultTopology.xml`
  Defines the default Topology module instance that is inserted into a portal view when using the `[Add]` button in the SIP user interface. This file is specified in the `defaultConfigXML` attribute in the `OVRegTopology.xml` file. (See the `/conf/.../views/OVTopology.dtd` file for more information.)
- `/conf/.../NM/nmConfig.xml`
  This file contains the list of all NNM management stations with which SIP is allowed to communicate. You must specify whether or not the Topology module is allowed to request data. You must provide information about which ports are being used by NNM processes to communicate topology data on each management station. See `nmConfig.dtd` and `nmConfig.xml` for more information.
- `/conf/.../topology/topologyConfig.xml`
  The settings in this file affect all Topology modules in all portal views (see the `topologyConfig.dtd` file for more information). Settings include:
  - — The maximum number of retry attempts allowed when accessing a submap on an NNM management station.
  - — The polling frequency (`symbolFetchRateInMin` attribute) for monitoring symbol registration file changes on NNM management stations.
  - — The default height/width settings of submaps.

  - — The `defaultFilter` attribute allows you to ignore filtering that has been defined in the customer model and management data information for the current role. If this attribute is set to NO, all data that is visible within NNM is displayed on the submaps in the SIP portal view. See the SIP *Administrator Guide* (`Administrator_Guide.pdf`) for information about customer models, management data filters, and roles.
  - — The `filterConnSymbols` attribute allows you to control which connection lines are shown on the SIP submaps.
  - — The `loadTransientSubmaps` attribute toggles drill-down access to *transient* submaps (those generated on-demand in NNM), as opposed to only allowing drill-down access to *persistent* submaps (those stored in RAM on the NNM management station) or transient submaps that are currently displayed.

- `/conf/.../views/`*`PortalView`*`.xml`
  Place the actual instances of the Topology module within each *`PortalView.xml`* file. In each module instance, you specify the NNM management station(s) and submap(s), whether or not drill-down is available, and whether symbol status colors display through the SIP portal view, or all symbols display as Administrative-Unmanaged (cream color by default). If desired, you can also over-ride all filtering and display the submap exactly as it appears on the NNM management station. (See the `PortalView.dtd` and `OVTopology.dtd` files for more information.)

The following topics are covered in the remainder of this chapter:

Registering The Topology Module

The Topology module must be registered with SIP.

This section focuses on the following two files:

- `/registration/OVRegTopology.xml`
- `/registration/defaults/OVDefaultTopology.xml`

See the `OVModuleRegistration.dtd` file and the SIP *Administrator Guide* (`Administrator_Guide.pdf`) for more information.

Editing Access

The following attribute in the `OVRegTopology.xml` file is automatically enabled or disabled based upon the current SIP portal user's Role *editing permissions*:

- `edit`
  This attribute is set to *yes*. When a user with "Editing" permissions displays a Topology module in the SIP portal, an edit button appears in the titlebar of the Topology module. This button provides access to limited editing functions: add submaps, delete submaps, change the order of submaps, change status color choices, and set or disable drill-down.

For more information about roles and editing permissions, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`).

The [Add] Button

The following attributes in the `OVRegTopology.xml` file are automatically enabled or disabled based upon the current SIP portal user's Role *editing permissions*:

- `add`
  Set this attribute to *yes*. The default Topology module appears in the list of available modules when a user with "Editing" permissions uses the [Add] button at the bottom of the SIP portal window.
- `defaultConfigXML`
  This attribute specifies which Topology module appears when a user

with "Editing" permissions uses the [Add] button in a SIP portal to insert a Topology module. The `OVDefaultTopology.xml`file contains the default Topology module. Specify any file that you want, or modify the Topology module within the current file to meet your needs. See the comments in the `/conf/.../views/OVTopology.dtd` file for more information.

For more information about roles and editing permissions, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`).

The [?] (Help) Button

The following attribute in the `OVRegTopology.xml` file controls the default behavior of the [?] button in the SIP portal:

- `help`
  This attribute specifies the default help topic (html file) that appears when a user clicks the [?] button on a Topology module: `/OvSipDocs/C/help/NNM/mapsView.html`. You can specify any html file that you want.

You can override the default topic and provide a customized topic on a module-by-module basis.

TIP

To specify a more specific help file on a module-by-module basis, add a `help` attribute to the `ModuleInstance` element. For more information, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`), “Creating Customized Help Topics for Supplied Modules.”

Filtering Possibilities for the Topology Module

The Effect of the Customer Model and MgmtData Filter

By default, the Topology module is influenced by the `MgmtData` filter you defined in the customer model configuration. Submap symbols for objects that are an IP node, IP interface, or non-IP interface (objects that pass the `MgmtData` filter) are displayed on the submaps within the Topology module in the SIP portal view. Other submap symbols, such as IP networks, segments, and container objects are only displayed in SIP if they are connected to an object that has passed the `MgmtData` filter.

The Topology module gives you the option of ignoring the `MgmtData` filter and displaying NNM submaps exactly as they appear on the NNM management station.

Filter Settings for All Topology Modules

There are two attributes in the `topologyConfig.xml` file used to control filtering for all Topology modules:

- `defaultFilter`
  If set to *yes*, the NNM submaps are filtered according to the settings in the applicable `MgmtData` filter for the particular customer.

  If set to *no*, the `MgmtData` filter is ignored, and the SIP submaps include all symbols that appear on the NNM management station.

- `filterConSymbols`
  If set to *yes*, means that only those interfaces specifically listed in any applicable `InterfaceList` element in the customer model, and passed through the `MgmtData` filters in the user role file, are displayed as connective lines in the SIP submap.

  If set to *no*, means that connective lines representing interfaces are displayed in SIP if the node to which they are connected passes the `MgmtData` filter, irrespective of any limitations specified in the `InterfaceList`.

If you make any changes to the `topologyConfig.xml` file, you must do the following:

WindowsNT/2000:
From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

UNIX:

As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

Stop on HP-UX: **`/sbin/init.d/ovsip stop`**
Start on HP-UX: **`/sbin/init.d/ovsip start`**

Stop on Solaris: **`/etc/init.d/ovsip stop`**
Start on Solaris: **`/etc/init.d/ovsip start`**

Bypassing Filtering for Specific Topology Modules

The following attribute setting enables you to optionally bypass filtering for individual submaps (see the `OVTopology.dtd` file for more information). This attribute is entered into a specific `Submap` element in a specific Topology module in a specific `PortalView.xml` file:

- `filter`
  If set to *yes*, this NNM submap is filtered according to the settings in the applicable `MgmtData` filter for the particular customer.

  If set to *no*, the `MgmtData` filter is ignored, and the SIP submap includes all symbols that appear on the NNM management station.

Establishing Global Settings for All Topology Modules

This section focuses on the following two files:

- `/conf/share/modules/topology/topologyConfig.xml`
- `/conf/share/modules/NM/nmConfig.xml`

If you make changes to either of these files, you must do the following:

WindowsNT/2000:
From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

UNIX:

As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

Stop on HP-UX: `/sbin/init.d/ovsip stop`
Start on HP-UX: `/sbin/init.d/ovsip start`

Stop on Solaris: `/etc/init.d/ovsip stop`
Start on Solaris: `/etc/init.d/ovsip start`

topologyConfig.xml/dtd

There are two ways to use the Topology module. You can display submaps from NNM management stations (or collection stations) or you can display a GIF file. The `topologyConfig.xml` file sets default settings for both of these, such as `width`/`height` of the displayed images. This file also contains attribute settings that control how the Topology modules interact with NNM. In addition to the filtering attributes discussed in “Filter Settings for All Topology Modules” on page 103, the following attributes control the frequency with which SIP Topology modules request updated information from NNM management stations:

- `numMapRetries`
  SIP starts checking for a specified map at port 3700. If a map is not running, SIP increments the counter and checks on the next port. If a

map was running on 3700, but not the desired map, SIP resets the counter to 0 and checks the next port. Once the counter equals `numMapRetries`, SIP quits searching for the requested map.

- `symbolFetchRateInMin`
  Sets the frequency (in minutes) with which SIP contacts NNM management stations to check for changes in symbol registration files and gathers the symbol GIF images.
- `defaultWidth & Height`
  Sets the dimensions of submaps when the dimension is not specified by the submap element in the Topology module.
- `drillDownWidth & Height`
  Sets the dimensions of submaps accessed through drill-down behavior.
- `loadTransientSubmaps`
  Toggles drill-down access to *transient* submaps (those generated on-demand in NNM), as opposed to only allowing drill-down access to *persistent* submaps (those stored in RAM on the NNM management station) or transient submaps that are currently displayed.

See the `topologyConfig.dtd` and `topologyConfig.xml` for more information.

nmConfig.xml/dtd

Only those NNM management stations or collection stations that are configured to do so in the `nmConfig.xml` file are allowed to provide submap information to SIP. See `nmConfig.xml` and `nmConfig.dtd` for more information. See also the `Configuring_NNM.pdf` file. The bold settings in the following example must be set before the Topology module works:

```
<NNMStation
     hostname="hostname or IP address of the NNM system"
     snmpDataSource="get SNMP data collection data?: yes or no"
     alarmsDataSource="get NNM alarms?: yes or no"
     symbolRegSource="get OVW symbol registration information?: yes or no"
     webSrvPort="NNM web server port: usually 80 for NT; 8880 for Unix"
     ovwdbPort="ovwdb port: usually 9999 for NNM 6.1; 2447 for NNM 6.2"
     ovAlarmSrvPort="ovalarmsrv port: 2345 for NNM 6.1; 2953 for NNM 6.2"
     encoding="if NNM is using other than English"
/>
```

The `symbolRegSource` attribute controls which NNM management stations are polled for changes to the symbol registration files. If you use a standard symbol set on all NNM management stations, you may want to set `symbolRegSource` attribute to `"yes"` for only one NNM management station, in order to keep network traffic to a minimum.

When the NNM management station is running in a language other than English, the `encoding` attribute specifies the code set from which SIP needs to convert data for display. NNM map data is translated into UTF-8 characters by the portal. See the "Internationalization" section of the SIP *Administrator Guide* (`Administrator_Guide.pdf`) for additional instructions about using this attribute.

NOTE

Verify that you are consistent in your usage of *either* hostname or IP address when specifying the NNM management station in the `nmConfig.xml` file and any Topology module instances.

Steps on the NNM Management Station

Each desired NNM map must be open on the NNM management station before submaps can be displayed in the SIP portal view. Only submaps currently displayed on the NNM management station or *persistent* submaps (those stored in memory on the NNM management station) can be displayed through SIP.

TIP

Check the following file for information about running NNM in a virtual window so that you don't have to keep every SIP map open in an `ovw` session: `/htdocs/WhitePapers/VirtualWindow-NNM.html`

To prepare the NNM maps:

1. Open NNM (ovw) on the NNM management station containing the map you wish to display in a SIP portal view:

   Windows NT/2000:

   - Start the NNM services (if necessary) by clicking `Start:Programs->HP OpenView-> Network Node Manager Admin->NNM Services-Start`.
   - Start the NNM interface by clicking `Start:Programs-> HP OpenView->Network Node Manager`.

   UNIX:

   - To start the NNM background processes, log in as `root` and type:

     `/opt/OV/bin/ovstart -c`

   - To start the NNM interface, type:

     `/opt/OV/bin/ovw`

2. Open the desired map on the NNM management station.
3. Ensure that the desired submap is displayed or set to *persistent* (stored in RAM, not *transient* -- generated upon request). To check or change persistence, do one of the following:

   - Configure the IP Map application to enable the on-demand level:

Windows NT/2000: `Map:Properties`. From the `Applications` tab, double click on IP Map and select an `On-Demand` level.

UNIX: `Map:Properties`. Select IP Map, click the [`Configure For This Map`] button, and select an `On-Demand` level.

- Make the individual submap persistent:

  Windows NT/2000:`Map:Submap:Properties`. From the `View` tab, select the `Persistent` check box.

  UNIX: `Map:Submap:Make the Submap Persistent`.

4. Create and customize any desired maps and submaps. (For information about map customization, see the NNM manual *Managing Your Network with NNM*.)

   Consider creating a few general purpose submaps, and using SIP filters to display only the information that is important for a specific customer.

5. A submap's background graphic, if any, can be automatically displayed in SIP. The graphic must be in either JPEG or GIF format, and the graphic must be placed in the following location on the NNM management station:

   - *Windows NT/2000*: `<NNM_install_dir>\backgrounds\*`
   - *UNIX*: `/usr/OV/backgrounds/*`

   To add a background graphic to a submap:

   a. On the NNM management station, open the NNM submap to which you wish to add a background graphic.

   b. Select `Map`: `Submap->Properties`.

      Only Windows NT/2000: click the `View` tab.

   c. In the `Background Graphics` list, select the graphic you want to apply to the current submap, then click the `[OK]` button.

   For more detailed information, see the NNM's online help or the *Managing Your Network with NNM* manual.

6. If the NNM management station is restarted, you must open each NNM session to enable display of the submaps in the SIP portal views.

Editing Topology Modules

Modify the Topology module by:

- Adding an Topology Module to a Portal View
- Choosing Submaps to Be Displayed
- Presenting Topology Submaps from Multiple NNM Management Stations
- Changing the Display Order of Submaps
- Removing Submaps from a Topology Module
- Showing/Hiding Status Information on Submaps
- Controlling Drill-Down through Submaps
- Changing the Size of a Submap
- Displaying a GIF File Instead of an NNM Submap

Adding an Topology Module to a Portal View

Two ways exist for adding Topology modules to a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *PortalView*.xml file you want to customize.
2. Navigate to the appropriate Tab.
3. At the bottom of the window, select Topology and click [Add].
4. Save the changes and return to the main portal page; click the [OK] button.

TIP

If you want to create and add a different instance of a module to the list of available modules, see the SIP *Administrator Guide* (Administrator_Guide.pdf)

Through the Portal View File

Modules can be added and configured by directly editing a `PortalView.xml` file. Modules are wrapped in the `ModuleInstance` element. The `ModuleInstance id` must be unique among all module instances in the portal view file. For information about the `ModuleInstance` element, see the SIP *Administrator Guide* (`Administrator_Guide.pdf`), "Designing Portal Views."

Follow the directions in the SIP *Administrator Guide*.

Refer to the `OVTopology.dtd` file for more information.

You can copy and paste the contents of the following file into your *`PortalView`*`.xml` file as a starting point:

- `OVDefaultTopology.xml`

Choosing Submaps to Be Displayed

Two ways exist for choosing the submaps presented in a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

To define the list of submaps displayed for a customer:

1. Log into the *`PortalView`*`.xml` file you want to customize.
2. Navigate to the appropriate Tab.

   If necessary, navigate to the `[Add]` button. Select Topology and click `[Add]`.
3. On the title bar of the Topology module, click the `[Edit]` button.
4. On the `Topology - Edit` page, in the `Management Stations` list, select the name of the NNM management station that has the submap you wish to access. The list contains all NNM management stations listed in the `nmConfig.xml` file.
5. In the `Map` field, type the name of the map, then click the `[List Submaps]` button.

   NOTE: Submaps must be currently displayed on the NNM management station or configured as *persistent* (not *transient*) within

NNM before they are available for display in the SIP portal. This means that the submap must be stored in RAM on the NNM management station and not generated on-the-fly upon request.

6. From the `Available Submaps` list, select a submap (if duplicate submap names occur in the list, verify the path displayed after the submap name).
7. Click the `[Add]` button. The submap name is moved to the `Submaps to Display` list and added to the bottom of the `Displayed Submaps` list.
8. In the `Displayed Submaps` list, select the submap name and use the `[Up]` and `[Down]` buttons to navigate the new submap into the correct display location.
9. Save the changes and return to the main portal page; click the `[OK]` button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *`PortalView`*`.xml` file, which is stored in the following directory:

   Windows NT/2000: `<`*`SIP_install_dir`*`>\conf\share\views\`
   UNIX: `/etc/opt/OV/SIP/conf/share/views/`

2. To add a submap to the Topology module instance, create a new line within the `TopologyMap` element, as follows (if you do this through the SIP user interface, the path to the submap is automatically determined):

```
<TopologyMap showStatus="yes" drillDown="yes">
    <Submap href="ovw://NNMhostName/mapName/submap">
    <Submap href="ovw:/NNMhostName/mapName/submap">
</TopologyMap>
```

NOTE Verify that you are consistent in your usage of *either* hostname or IP address when specifying the NNM management station in the `nmConfig.xml` file and any Topology module instances.

See the `OVDefaultTopology.xml` file for an example.

If you prefer to display a GIF file that looks like an NNM submap, create a new line as follows:
`<Submap href="http://URLforTheImage>`

3. Close and save the *`PortalView`*`.xml` file.
4. In a browser, log into the portal the portal to verify that the submaps appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Presenting Topology Submaps from Multiple NNM Management Stations

Two ways exist for choosing submaps from multiple NNM management stations:

- Through the User Interface
- Through the Portal View File

Through the User Interface

You can choose to display submaps from multiple NNM management or NNM collection stations.

1. Log into the *PortalView*.xml file you want to change.
2. Navigate to the appropriate Tab.

   If necessary, navigate to the [Add] button. Select Topology and click [Add].
3. On the title bar of the Topology module, click the [Edit] button.
4. On the Topology - Edit page, in the Management Stations list, select the name of the NNM management station that has the submap you wish to access. The list contains all NNM management stations listed in the nmConfig.xml file.
5. In the Map field, type the name of the map, then click the [List Submaps] button.

   NOTE: Submaps must be currently displayed on the NNM management station or configured as *persistent* (not *transient*) within NNM before they are available for display in the SIP portal. This means that the submap must be stored in RAM on the NNM management station and not generated on-the-fly upon request.
6. From the Available Submaps list, select a submap (if duplicate submap names occur in the list, verify the path displayed after the submap name).
7. Click the [Add] button. The submap name is moved to the Submaps to Display list and added to the bottom of the Displayed Submaps list.
8. In the Displayed Submaps list, select the submap name and use the [Up] and [Down] buttons to navigate the new submap into the correct display location.
9. Return to step 4 and select the next NNM management station.
10. Save the changes and return to the main portal page; click the [OK] button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to

revert to the previous version of the file.

1. In an ASCII or XML editor, open the *PortalView*.xml file, which is stored in the following directory:

   Windows NT/2000: <*SIP_install_dir*>\conf\share\views\
   UNIX: /etc/opt/OV/SIP/conf/share/views/

2. Both SIP and any NNM management station that SIP should contact, must be appropriately configured. See "Establishing Communication Between NNM and SIP" in the SIP *Configuring NNM* manual (Configuring_NNM.pdf) for more information.

3. To add a submap to the Topology module instance, create a new line within the TopologyMap element, as follows (if you do this through the SIP user interface, the path to the submap is automatically determined):

   ```
   <TopologyMap showStatus="yes" drillDown="yes">
       <Submap href="ovw://NNMhostName/mapName/submap">
       <Submap href="ovw://NNMhostName/mapName/submap">
   </TopologyMap>
   ```

NOTE Verify that you are consistent in your usage of *either* hostname or IP address when specifying the NNM management station in the nmConfig.xml file and any Topology module instances.

   See the OVDefaultTopology.xml file for an example.

   If you prefer to display a GIF file that looks like an NNM submap, create a new line as follows:
   <Submap href="http://*URLforTheImage*">

4. Close and save the *PortalView*.xml file.

5. In a browser, log into the portal the portal to verify that the submaps appear as desired.

NOTE After you make modifications to XML files, validate the syntax. Provided with SIP is the command xmlvalidate, which checks whether the XML file is both well-formed and valid. xmlvalidate uses the same XML parser as SIP, so if the file passes xmlvalidate, it will work with SIP. The correct usage of this command is: xmlvalidate <xml filename>.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Changing the Display Order of Submaps

Two ways exist for changing the order of submaps presented in a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *`PortalView`*`.xml` file you want to change.
2. Navigate to the appropriate Tab.

   If necessary, navigate to the `[Add]` button. Select Topology and click `[Add]`.
3. On the title bar of the Topology module, click the `[Edit]` button.
4. On the `Topology - Edit` page, in the `Displayed Submaps` list, select the submap name and use the `[Up]` and `[Down]` buttons to navigate the submap into the correct display location.
5. Return to step 4 and select the next submap you wish to move.
6. Save the changes and return to the main portal page; click the `[OK]` button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *PortalView*.xml file, which is stored in the following directory:

   Windows NT/2000: <*SIP_install_dir*>\conf\share\views\
   UNIX: /etc/opt/OV/SIP/conf/share/views/

2. To change the order of submaps in the Topology module instance, select a Submap element line within the TopologyMap element, for example:

   ```
   <TopologyMap showStatus="yes" drillDown="yes">
       <Submap href="ovw://NNMhostName/mapName/submap">
       <Submap href="ovw://NNMhostName/mapName/submap">
   </TopologyMap>
   ```

3. Cut and paste the Submap element to the new location
4. Repeat this procedure until the submaps are in the desired order.
5. Close and save the *PortalView*.xml file.
6. In a browser, log into the portal the portal to verify that the submaps appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command xmlvalidate, which checks whether the XML file is both well-formed and valid. xmlvalidate uses the same XML parser as SIP, so if the file passes xmlvalidate, it will work with SIP. The correct usage of this command is: xmlvalidate <xml filename>.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: %SIP_HOME%\bin
UNIX: /opt/OV/SIP/bin

If the output of the xmlvalidate command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer

which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Removing Submaps from a Topology Module

Two ways exist for removing the submaps from a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *`PortalView`*`.xml` file you want to change.
2. Navigate to the appropriate Tab.

   If necessary, navigate to the `[Add]` button. Select Topology and click `[Add]`.
3. On the title bar of the Topology module, click the `[Edit]` button.
4. On the `Topology - Edit` page, in the `Displayed Submaps` list, select the submap name and click the `[Delete]` button to remove the submap.
5. Return to step 4 and select the next submap you wish to remove.
6. Save the changes and return to the main portal page; click the `[OK]` button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *`PortalView`*`.xml` file, which is stored in the following directory:

   Windows NT/2000: `<`*`SIP_install_dir`*`>\conf\share\views\`
   UNIX: `/etc/opt/OV/SIP/conf/share/views/`
2. To remove a submap from the Topology module instance, delete the `Submap` element line within the `TopologyMap` element, as follows (if

you have more than one Topology module in this xml file, make sure you have to the right one):
`<Submap href="ovw://NNMhostName/mapName/submap">`

Or

`<Submap href="http://URLforTheImage>`

3. Close and save the *PortalView*.xml file.
4. In a browser, log into the portal the portal to verify that the submaps appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Showing/Hiding Status Information on Submaps

Two ways exist for controlling symbol status in the submaps presented in a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *PortalView*.xml file you want to change.
2. Navigate to the appropriate Tab.

   If necessary, navigate to the [Add] button. Select Topology and click [Add].
3. On the title bar of the Topology module, click the [Edit] button.
4. On the Topology - Edit page, toggle the Show Status in submap check box. All displayed submaps in this Topology module are affected by this symbol status setting:

   - If selected, all symbols assume the NNM *administrative status* of "unmanaged" (cream colored by default). All connection lines remain black.
   - If deselected, all symbols and connection lines display their current status color from NNM.
5. Save the changes and return to the main portal page; click the [OK] button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *PortalView*.xml file, which is stored in the following directory:

   Windows NT/2000: <*SIP_install_dir*>\conf\share\views\
   UNIX: /etc/opt/OV/SIP/conf/share/views/
2. To show/hide the current status colors of NNM objects on SIP submaps, locate the TopologyMap element (if you have more than one Topology module in this xml file, make sure you have to the right one).
3. Make your change to the showStatus attribute; for example:

   <TopologyMap **showStatus="yes"** drillDown="no">

   "yes" = NNM's current symbol status colors display through the SIP portal view.

"no" = all symbols displayed in SIP assume NNM's administrative-Unmanaged status color (cream color by default) .

4. Close and save the *PortalView*.xml file.
5. In a browser, log into the portal the portal to verify that the submaps appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command xmlvalidate, which checks whether the XML file is both well-formed and valid. xmlvalidate uses the same XML parser as SIP, so if the file passes xmlvalidate, it will work with SIP. The correct usage of this command is: xmlvalidate <xml filename>.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: %SIP_HOME%\bin
UNIX: /opt/OV/SIP/bin

If the output of the xmlvalidate command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at www.xmlspy.com.

Controlling Drill-Down through Submaps

Two ways exist for controlling drill-down behavior in the submaps presented in a portal view:

- Through the User Interface
- Through the Portal View File

Through the User Interface

1. Log into the *PortalView*.xml file you want to change.
2. On the title bar of the Topology module, click the [Edit] button.

3. On the `Topology - Edit` page, toggle the `Drill Down in submaps` check box. All displayed submaps in this Topology module are affected by this symbol status setting:

   - If selected, all submaps allow drill-down access through the NNM hierarchy.

     The submaps that can be accessed through drill-down are controlled by the global settings in the `topologyConfig.xml` file. See "Establishing Global Settings for All Topology Modules" on page 105 for more information.

   - If deselected, none of the displayed submaps provide drill-down access.

4. Save the changes and return to the main portal page; click the `[OK]` button.

Through the Portal View File

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *`PortalView`*`.xml` file, which is stored in the following directory:

   Windows NT/2000: *`<SIP_install_dir>`*`\conf\share\views\`
   UNIX: `/etc/opt/OV/SIP/conf/share/views/`

2. To enable or disable drill-down behavior for SIP submaps, locate the `TopologyMap` element (if you have more than one Topology module in this xml file, make sure you have to the right one).

3. Make your change to the `drillDown` attribute; for example:

   `<TopologyMap showStatus="yes"` **`drillDown="yes"`**`>`

   `"yes"` = all submaps allow drill-down access through the NNM hierarchy. The submaps that can be accessed through drill-down are controlled by the global settings in the `topologyConfig.xml` file. See the `topologyConfig.dtd` file"Establishing Global Settings for All Topology Modules" on page 105 for more information.

   `"no"` = none of the displayed submaps provide drill-down access.

4. Close and save the *`PortalView`*`.xml` file.

5. In a browser, log into the portal the portal to verify that the submaps appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Changing the Size of a Submap

Default width and height values (550 and 300, respectively) are assigned in the `topologyConfig.xml` file. If you want to override these values, you can directly edit the *`PortalView`*`.xml` file.

As a general rule, make a backup of *`PortalView`*`.xml` files before you change them. If you edit the *`PortalView`*`.xml` file and get incorrect XML syntax, you may want the ability to revert to the previous version of the file.

1. In an ASCII or XML editor, open the *`PortalView`*`.xml` file, which is stored in the following directory:

   Windows NT/2000: `<SIP_install_dir>\conf\share\views\`
   UNIX: `/etc/opt/OV/SIP/conf/share/views/`

2. To change the size of a submap, locate the `TopologyMap` element (if you have more than one Topology module in this xml file, make sure you have to the right one).
3. Select a `Submap` element line within the `TopologyMap` element, for example:
   `<Submap href="ovw://NNMhostName/mapName/submap">`
4. Add the width and height attributes, for example:

```
<Submap href="ovw://NNMhostName/mapName/submap" width=550 height=300/>
```

   Add (or change) the `width` and `height` attributes.

5. Close and save the *`PortalView`*`.xml` file.
6. In a browser, log into the portal the portal to verify that the submaps appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command `xmlvalidate`, which checks whether the XML file is both well-formed and valid. `xmlvalidate` uses the same XML parser as SIP, so if the file passes `xmlvalidate`, it will work with SIP. The correct usage of this command is: `xmlvalidate <xml filename>`.

An XML file is "well-formed" if it conforms to a minimal set of rules defined for all XML documents. It is "valid" if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: `%SIP_HOME%\bin`
UNIX: `/opt/OV/SIP/bin`

If the output of the `xmlvalidate` command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at `www.xmlspy.com`.

Displaying a GIF File Instead of an NNM Submap

Make a backup of XML files before you customize them. If you edit the XML file and get incorrect XML syntax, you may want the ability to

revert to the previous version of the file.

1. In an ASCII or XML editor, open the *PortalView*.xml file, which is stored in the following directory:

   Windows NT/2000: <*SIP_install_dir*>\conf\share\views\
   UNIX: /etc/opt/OV/SIP/conf/share/views/

2. To display a bitmap in the Topology module, rather than an NNM submap, create a new line within the TopologyMap element, as follows (you can include bitmaps and submaps within the same Topology module instance):

   ```
   <TopologyMap showStatus="yes" drillDown="yes">
       <Submap href="http://URLforTheImage>
       <Submap href="http://URLforTheImage>
   </TopologyMap>
   ```

3. Close and save the *PortalView*.xml file.

4. In a browser, log into the portal the portal to verify that the submaps appear as desired.

NOTE

After you make modifications to XML files, validate the syntax. Provided with SIP is the command xmlvalidate, which checks whether the XML file is both well-formed and valid. xmlvalidate uses the same XML parser as SIP, so if the file passes xmlvalidate, it will work with SIP. The correct usage of this command is: xmlvalidate <*xml filename*>.

An XML file is “well-formed” if it conforms to a minimal set of rules defined for all XML documents. It is “valid” if it conforms to the DTD listed at the beginning of the XML file.

Make sure the following has been added to your PATH variable:

Windows NT/2000: %SIP_HOME%\bin
UNIX: /opt/OV/SIP/bin

If the output of the xmlvalidate command indicates a problem but does not fully describe it, you can open the XML file in Internet Explorer which sometimes provides a clearer description of the problem.

As an alternative, you can find a XML validation tool for Windows NT at www.xmlspy.com.

Location of Relevant Files

Table 4-1 Topology Module Files on the SIP Server

File Name	Windows NT/2000 Location <*SIP_install_dir*>/SIP/....	UNIX Location etc/opt/OV/SIP/....
nmConfig.dtd	conf/share/modules/NM/	conf/share/modules/NM/
nmConfig.xml	conf/share/modules/NM/	conf/share/modules/NM/
OVModuleRegistration.dtd	registration/	registration/
OVRegTopology.xml	registration/	registration/
topologyConfig.dtd	conf/share/modules/topology/	conf/share/modules/topology/
topologyConfig.xml	conf/share/modules/topology/	conf/share/modules/topology/
OVTopology.dtd	conf/share/views/	conf/share/views/
OVDefaultTopology.xml	registration/defaults/	registration/defaults/
PortalView.xml	conf/share/views	conf/share/views

5 Troubleshooting

General

"Tab pages containing any NNM module are blank within the SIP portal"

Symptom:

The tab page is empty. The progress bar looks like the page is partially loaded, but never finishes.

Cause A:

One of the NNM management stations (that SIP gathers data from) is in a paused state, for example during the early phase of an NNM backup procedure. This causes a problem for SIP because the NNM `ovwdb` process must be stopped for a short time during the backup procedure to ensure that the NNM databases are not corrupted.

Solution A:

Wait for NNM's backup to proceed beyond the `ovpause` state. The modules display when the NNM management station issues an `ovresume` command. If the browser timeout limit is exceeded while you are waiting, you must press `[Refresh]` to display the modules.

Or on the NNM management station, issue the `ovresume` command, if appropriate to do so without disturbing the regularly scheduled NNM backup procedure.

Alarms Module

The portal fails to display any alarms data

Possible Cause A:

Alarm categories can be configured to filter alarms in a number of ways. If you implement a `CapabilityFilter` within a `MgmtData` filter definition or within an alarm category's `NodeSelection` filter definition, the Alarms module has a dependency upon NNM's `ovwdb` process.

The portal may not be communicating with `ovwdb`.

Solution:

Restart `ovw` on the target system via `ovstart`.

Possible Cause B:

The portal may not be communicating with `ovalarmsrv`.

Solution:

Use the `ovstatus` command to validate the status of `ovalarmsrv`.

Restart `ovalarmsrv`, if necessary, by issuing the `ovstart` command.

Try to communicate directly with `ovalarmsrv`:

1. Enter: `telnet <NNMStationName> <ovAlarmsSrvPort>`.
2. In the telnet window, enter `0:0:CATEGORIES:TestUser`.

   If you get a response, it's up and running.
3. In the telnet window, enter `6` to end communications.

Possible Cause C:

Invalid port configured for `ovalarmsrv`.

Solution:

1. If the port number configured for an NNM system is invalid, the alarm module will not be able to obtain alarms for display from this system. Check which port each NNM station is communicating with. The port that the bits respond on depends on the entries in the `services` file.

On UNIX, the `services` file resides in `/etc/services`. `Ovalarmsrv` has two entries: `ovalarmsrv` and `ovalarmsrv_cmd`. The value that is set in the file determines which port `ovalarmsrv` runs on. (The same is true of `ovwdb`.)

On Windows NT/2000, the `services` file resides in `WINNT\system32\drivers\etc\services`.

2. Modify the entry or entries in `nmConfig.xml` as necessary to match what you find.
3. *WindowsNT/2000*:
   From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

   UNIX:

   As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **`/sbin/init.d/ovsip stop`**
   Start on HP-UX: **`/sbin/init.d/ovsip start`**

   Stop on Solaris: **`/etc/init.d/ovsip stop`**
   Start on Solaris: **`/etc/init.d/ovsip start`**
4. Changes to portal view files take effect when you display or refresh the portal view.
5. If you still do not see data, verify that you are communicating with `ovalarmsrv` on each NNM station listed in `nmconfig.xml` by completing steps 1-3 in the solution to Possible Cause B.

Possible Cause D:

Specified NNM stations do not match those in `nmConfig.xml`.

Solution:

Resolve the differences by editing the alarm category definition file and the `nmConfig.xml` file.

Possible Cause E:

Alarms categories defined in the `portalview.xml` files don't match alarm categories in the `NmAlarmCatsIndex.xml` file.

Solution:

Make sure they match.

Possible Cause F:

Base categories used in alarm category definition file not valid for the given NNM station.

Solution:

Make sure you are using valid NNM base categories for the stations you are connecting to.

Possible Cause G:

No data passed filters.

Solution:

Check the management data filters defined for this role.

Check any substring match, OlderThanXMins sevs, acks, or node selection filters defined for this category.

Possible Cause H:

No NNM stations configured in `nmConfig.xml`.

Solution:

Make sure stations are listed.

Make sure the correct `ovAlarmSrvPort` is listed.

Make sure `alarmsDataSource` is set to yes.

Possible Cause I:

Timeout values are too short.

Solution I:

See “The portal fails to display a specific alarm category.”

The portal fails to display a specific alarm category

Possible Cause A:

Timeout values are too short.

Solution A:

1. Modify the following attributes in the `NmAlarmConfig.xml` file. (Note: Each time you modify these attributes, you will need to complete steps 2 and 3 to see your changes.) More information about these attributes is available in the comments within the `NmAlarmConfig.dtd` file:

   - `connTimeout` — The number of seconds (zero or greater) to pause after each socket connection is opened.
   - `addSyncTime` — The number of seconds (zero or greater) to add to `connTimeout` when making a synchronous call to get data from the `ovalarmsrv` on each NNM management station.
   - `socketTimeout` — The number of seconds (zero or greater) to wait for a socket connection to be made.
   - `responseTimeout` — The number of seconds (zero or greater) to wait each time for any response (protocol or data) from `ovalarmsrv`.
   - `maxWaitTime` — The maximum number of seconds (zero or greater) to wait for a data response from `ovalarmsrv`.

2. *WindowsNT/2000*:
   From the `Control Panel`, select `Services`. Stop and then restart `Tomcat`. Alternatively, you can use the command line: **`net stop tomcat`** and **`net start tomcat`**.

   UNIX:

   As `root`, stop and restart the web server and servlet engine by running the following. (The `DISPLAY` variable must be configured prior to restarting the webserver and servlet engine.)

   Stop on HP-UX: **`/sbin/init.d/ovsip stop`**
   Start on HP-UX: **`/sbin/init.d/ovsip start`**

   Stop on Solaris: **`/etc/init.d/ovsip stop`**
   Start on Solaris: **`/etc/init.d/ovsip start`**

3. Changes to portal view files take effect when you display or refresh the portal view.

"Invalid XML" error message

Possible Cause:

Bad configuration file.

Solution:

Check the log file for parse errors, run the XML validator and make sure that you have a legitimate XML file. The problem could be in either the users configuration XML file or the configuration of the alarm display file.

The `portal_log` file is located in the following directory:

Windows NT/2000: `<SIP_install_dir>\log\sip.log`

UNIX: `/var/opt/OV/SIP/log/sip.log`

SNMP Data Collection

Data collection configuration did not get updated to reflect changes in gauge definitions or Customer Model configurations

Symptom:

The expected data collection did not happen for devices added to a network device health gauge by expanding the filters or adding or creating a new gauge. Or, the expected data collection did not happen after changing or adding a configuration for a particular organization in the Customer Model.

Possible Cause:

- Automatic data collector configuration may not have been enabled on the NNM stations.
- The `ovcolautoconf` program may not have run since you made the change. NNM's SNMP data collection configuration is updated by this program.
- The portal tab containing the Network Device Health module for this user/role has never been displayed.
- The portal tab containing the Network Device Health module for this user/role has not been displayed within the last 30 days. `ovcolautoconf` removes configuration entries for data that has not been requested within the last 30 days. (30 is the default and can be overwritten with the `-maxConfAge` option on `ovcolautoconf`.)
- SIP may not yet have sent its configuration requests to the NNM stations.
- Or, `ovcolautoconf` may be experiencing errors.

Solution:

1. Enable `autoDCConfig` if it is not already enabled. See the *Configuring_NNM.pdf* for information.
2. Display the portal tab containing Network Device Health.
3. Wait ten minutes.

4. Run `ovcolautoconf`. At the command prompt on the NNM systems, type: `ovcolautoconf -verbose`.(You may wish to run this command as a scheduled task.) Note that for UNIX `/opt/OV/bin` must be in your path.

   Collected data generally appears within a half hour of executing this command.

NOTE

Any network device that you want to be included in the SNMP data collection process must be a *managed* device within the NNM topology database, and NNM must know the correct SNMP GET community name for that device before the device will be included in the SNMP data collection process.

For information about the steps required when the Service Information Portal software requests SNMP data from NNM, see the *Configuring_NNM.pdf* and *Presenting_NNM_Data.pdf*. Verify that each step of the process is working correctly.

For other possible causes and solutions, see “Data not available” Error Message In Details Table under Network Device Health troubleshooting.

NNM Data Collector files of my SIP information are not being trimmed

Symptom:

NNM’s `snmpcollect` database is growing without bound.

Possible Cause:

No steps have been taken to trim the NNM `snmpCollect` database.

Solution:

See the NNM manual *Managing Your Network with NNM* or the SIP manual *Configuring_NNM.pdf* for how to trim data in the `snmpcollect` database.

Extraneous data collections are being gathered for network device health gauges

Symptom:

When I check the file `snmpRepPrev.conf`, there are entries for devices for which I do not want to collect data.

Possible Cause A:

Are you collecting more data than you need? Check the Customer Model definitions in `CustomerModel.xml`. Check your network device health filter specifications within each *`PortalView`*`.xml` file.

Solution:

As necessary, modify the `CustomerModel.xml` and/or the `NodeSelection` and `InterfaceSelections` within each *`PortalView`*`.xml` file.

Possible Cause B:

The unwanted entries may have been added at an earlier time, but due to SIP configuration changes, they are no longer needed.

Solution:

By default, `ovcolautoconf` removes configuration entries that have not been needed for 30 days. Run `ovcolautoconf`, using the `-maxConfAge` option if desired, to remove younger entries. See “ovcolautoconf.exe” on page 93 in the Network Device Health Gauge Module chapter.

xnmcollect -snmpColConfFile doesn't work

Symptom:

Windows NT/2000: When I try to edit the NNM data collection configuration for the network device health gauges by typing the following command, I get an error message and cannot edit the file:

```
xnmcollect -snmpColConfFile
<NNM_install_dir>\conf\ovcolautoconf\snmpRepAuto.templ
```

Possible Cause:

A patch is needed.

Solution:

You need to install NNM patch 00595 (or the superseding patch). The patch is available at the following website:

```
http://ovweb.external.hp.com/cpe/patches
```

Network Device Health Gauges

"Currently not configured" error message instead of gauge

Symptom:

No data is displayed in the gauge. The "Currently not configured" error message displays instead.

Possible Cause A:

No NNM stations are configured in `nnmConfig.xml`, or there are no NNM stations configured with the `snmpDataSource` attribute set to "yes."

Solution:

Make sure there is at least one NNM station entry in the `nmConfig.xml` with the `snmpDataSource` attribute set to "yes."

Possible Cause B:

The combination of the `MgmtData` filter and the gauge's `NodeSelection` or `InterfaceSelection` results in no filtering. In other words, all nodes/interfaces pass the filters. Computation on *all* nodes/interfaces is not supported. See the `sip.log` file for specific error messages.

The `sip.log` file is located in the following directory:

Windows NT/2000: `<SIP_inst_dir>\log\sip.log`

UNIX: `/var/opt/OV/SIP/log/sip.log`

Solution:

Limit the number of devices that pass the gauge's filters by doing one or more of the following (for more information about filters, see *Presenting_NNM_Data.pdf*):

- Narrow the `MgmtData` filter for this user role.
- In the *`PortalView`*`.xml` configuration file, narrow the gauge's `NodeSelection` or `InterfaceSelection` filter. To avoid having the same problem the next time you create a new *`PortalView`*`.xml` file, modify the gauge definitions in the `OVDefaultNetHealth.xml` file.

When you insert the Network Device Health module into a new portal for the first time, all gauges defined within the `OVDefaultNetHealth.xml` file are copied into the *`PortalView`*`.xml` file.

"Managed objects not found" error message instead of gauge

Symptom:

No data is displayed in the gauge. The "Managed objects not found" error message is displayed instead.

Possible Cause A:

The combination of the customer's `MgmtData` filter of the current user role and the gauge's `NodeSelection` or `InterfaceSelection` filter settings are so restrictive that no network devices can pass. An entry will be logged to `sip.log`, such as "No Nodes found for health summary category <*`category name`*>" or "No Interfaces found for health summary category <*`category name`*>".

This is most likely to occur with Key Device Health, CPE Health and Server Health.

The `sip.log` file is located in the following directory:

Windows/NT2000: <*`SIP_inst_dir`*>`\SIP\log\sip.log`

UNIX: `/var/opt/OV/SIP/log/sip.log`

Solution:

In NNM, select `Edit->Find->Object By Attribute` to determine if there are any devices with the specified capability set to TRUE (`isKeyDevice`, `isCPE`, `isServer`). If such nodes exist, do they pass the customer's `MgmtData` filter? If the desired capability is not set for one or more nodes, see *Managing Your Network with NNM* for information about how to set NNM object capabilities for the various network devices.

If necessary, modify the `MgmtData` filter and/or locate the gauge's <*Summary*> (definition in *`PortalView`*`.xml`) and modify the `NodeSelection` or `InterfaceSelection` filter. For more information, see *Presenting_NNM_Data.pdf*.

Possible Cause B:

An NNM station entry in `nmConfig.xml` is incorrectly specified.

Solution:

Verify that the `ovwdbPort` attributes specified in `nmConfig.xml` are correct. For more information, see the comments in `nmConfig.xml` or *Configuring_NNM.pdf*.

"Data currently unavailable" error message instead of gauges

Symptom:

No data is displayed in any gauge. The "Data unavailable" error message displays instead.

Possible Cause A:

The `sip.log` file contains a detailed message about the problem. There may be a syntax error in the `netHealthConfig.xml` file. (For example, the href syntax for a health Element may be invalid.)

Solution:

Check the `sip.log` file for a detailed message about the problem.

The `sip.log` file is located in the following directory:

Windows NT/2000: `<SIP_install_dir>\SIP\log\sip.log`

UNIX: `/var/opt/OV/SIP/log/sip.log`

Restore the `netHealthConfig.xml` file to its last working state or fix the syntax error identified in the `sip.log` file.

See the comments in `netHealthConfig.dtd`.

Possible Cause B:

The CGI program `getnnmdata.exe` may not be on the NNM station.

Solution:

If the NNM station is running version 6.1 of NNM, see *Configuring_NNM.pdf* for instructions on how to install CGI programs needed by SIP.

Possible Cause C:

The `hostname` or `webSrvPort` attributes in `nmConfig.xml` may be incorrectly specified.

Solution C:

Verify the `hostname` and `webSrvPort` attributes are correct. See the comments in `nmConfig.xml` or *Configuring_NNM.pdf* for more information.

Possible Cause D:

SIP 1.0 was uninstalled from one of your NNM management stations after configuring the NNM management station for SIP 2.0. Uninstalling SIP 1.0 uninstalls a library that is required by the SIP 2.0 CGIs. The library name is:

Windows NT/2000: `std312d.dll`

UNIX: `libstd12d.sl`

Solution D:

Check the `sip.log` for the following error message:

```
error   NetDevHealth    Thread-21       988216387619    Unable to successfully
invoke http://<hostname>:<port>/OvCgi/getnnmdata.exe -- 405:
java.io.FileNotFoundException: http://<hostname>:<port>/OvCgi/getnnmdata.exe
```

You can fix the problem by repeating the CGI installation process described in *Configuring_NNM.pdf*. This will reinstall SIP 2.0 CGI executables and the library that was removed. It will NOT overwrite the configuration files, `mibExprAuto.conf` and `snmpRepAuto.templ`, so any customizations you may have made are preserved.

"Data unavailable" error message in details table for all scores except Interface Status

Symptom:

"Data unavailable" appears instead of data in the detail tables.

Possible Cause A:

The NNM SNMP Data Collector may not be configured to collect the data needed by SIP.

Solution:

See "Data collection configuration did not get updated to reflect changes

in gauge definitions or Customer Model configurations" on page 87 and *Configuring_NNM.pdf*.

Possible Cause B:

NNM may not be able to contact the nodes in question via SNMP.

Solution:

In NNM, highlight the node, and select `Tools->SNMP MIB Browser` and walk the MIB2 interfaces group to see if the node is responding to SNMP requests. If it is not responding, there are several possibilities:

- The node may be down. Does the Interface Status column show "Down"? Does the node respond to ping?
- NNM may be using the wrong SNMP GET community string for the node. In NNM, select `Options->SNMP Configuration` to determine what community string NNM is using for the node. If you change one of these, while logged in as `root` or administrator, at the command prompt, type **`snmpCollect -C <nodename>`**
- The node's SNMP agent is not up or not responding.

Possible Cause C:

Network Node Manager is having problems with the SNMP data collection process.

Solution:

Run **`ovstatus-c snmpCollect`** on NNM to verify `snmpCollect` is running. See NNM log file `../log/snmpCol.trace` on the NNM system.

Possible Cause D:

By default, Service Information Portal gauges only use data up to 1 hour old. Perhaps the data is too old to be considered "near real-time" by network device health.

Solution:

To increase the acceptable age for SNMP data, increase the `maxAge` attribute in the `netHealthConfig.xml` file (the value represents minutes). This setting affects all gauges in all defined customer portals.

Possible Cause E:

Did you enable automatic data collection configuration on the NNM station? If so, did you edit the `snmpRepAuto.templ` file directly? If there is a syntax error in this file, the data collection process will fail.

Solution:

If the problem arose after you edited the `snmpRepAuto.templ` file, restore the `snmpRepAuto.templ` file to its last working state and following the directions in `snmpRepAuto.templ` when making changes.

"Data unavailable" error message on one row of details table (for a particular node or interface)

Symptom:

"Data unavailable" appears in the detail tables.

Possible Cause A:

NNM may not be able to contact the node in question via SNMP.

Solution:

For single MIB values, in NNM highlight the node, and select `Tools->SNMP MIB Browser`. Walk the MIB2 interfaces group to see if the node is responding to SNMP requests. For collections on MIB expressions, in NNM go to `Options->Data Collections & Thresholds:SNMP`, highlight the collection in question, and choose `Actions->Test SNMP`.

If it is not responding, there are several possibilities:

- The node may be down. Does the `Interface Status` column show `Down`? Does the node respond to ping?
- NNM may be using the wrong SNMP GET community string for the node. In NNM, select `Options->SNMP Configuration` to determine what community string NNM is using for the node. If you change one of the community strings and want to immediately attempt to reinitialize for a particular node (instead of waiting until the scheduled data collection check), while logged in as `root` or administrator, at the command prompt, type **`snmpCollect -C <nodename>`**
- The node's SNMP agent is not up or not responding.

Possible Cause B:

An SNMP agent patch may be required on the node in question.

Solution:

If the node is an HP-UX node, the column is `Interface % Utilization`, and the raw utilization value is greater than 100%, this is due to a known SNMP agent defect on HP-UX. The 11.0 patch for the HP-UX SNMP Agent software that fixes this problem is PHNE_21673 from the following web site:

`http://www.hp.com`, then click "HP Services & Support", Servers: UNIX

Note: When this agent defect is encountered, a warning is logged in the `sip.log` file: "Data value XYZ does not fall into any of the specified XML ranges."

Possible Cause C:

NNM may have incomplete network interface information.

Solution:

Check for valid IF Index values in NNM's topology database. In NNM, highlight the node, drill down into the node's Interface submap. Right click on an interface in question, and select `Interface Properties`. Examine the `Interface #` field. If this is blank or 0 (zero), Network Device Health is not able to retrieve SNMP data from this node. Such interface numbers values sometimes occur when NNM's discovery has not been allowed to complete for a node. Verify that NNM's `netmon` process is running. Is NNM's auto-discovery enabled? Are the node and interface *managed* within NNM?

Possible Cause D:

The node in question may not support one of the MIB variables used in computing that column value.

Solution:

The most common case of this is the `CPU Utilization` column in the first level of node drill down. This uses the Cisco MIB variable `cisco.local.lsystem.avgBusy5`, hence non-Cisco nodes will display the "Data unavailable" string for this column.

One approach to determining which MIB variable is unsupported is to let `snmpCollect` tell you what is wrong:

1. Toggle on `snmpCollect` tracing: `snmpCollect -T`
2. Toggle on `snmpCollect` verbose tracing: `snmpCollect -V`
3. Force a collection check on the node in question: `snmpCollect -C` *<nodename>*

4. Toggle off `snmpCollect` verbose tracing: `snmpCollect -V`
5. Toggle off `snmpCollect` tracing: `snmpCollect -T`
6. Examine `$OV_LOG/snmpCol.trace` for messages indicating why `snmpCollect` couldn't set up the collections for that node. (One possible message is "Expression stack underflow". This indicates a syntax error in the MIB expression as defined in `$OV_CONF/mibExpr.conf`. To correct this error, install the following NNM patch:

   HP-UX 11.0: `PHSS_23421`

   Windows NT/2000: `NNM_00674`

   Solaris: `PSOV_02882`

In general, you can do the following to determine which MIB variables are used in computing health column values:

1. Go to `netHealthConfig.xml`. Find the `Metric` whose Title matches the column title in question (for example, CPU Utilization).
2. Look at the last part of the href attribute of the Element to determine the NNM MIB expression/variable used. For example, `href="snmp://%item%[0]/p_cisco5minavgbusy"` indicates that the MIB expression `p_cisco5minavgbusy` is being used.
3. If a MIB expression (not a simple MIB variable) is being used, to determine which MIB variables are requested in the mathematical formula, open NNM and select `Options->Data Collection & Thresholds: SNMP`. Find the expression in the `MIB Objects Configured for Collection` list. Double-click on the entry. This will bring up a dialog box. Click on `[Describe]`:

   - **Direct NNM MIB expression**: shows the mathematical formula of MIB variables that the direct MIB expression is using.
   - **Indirect NNM MIB expression**: shows a list of possible direct MIB expressions in use. The actual direct MIB expression used will depend upon the attributes of the interface. To determine which direct MIB expression is being requested from a specific node, exit the `Description` dialog box, and in the `MIB Object Collection Summary` list click on the node in question and select `Actions->Test SNMP`. Note which direct MIB expression is being requested for each interface (for example, `IfHDplxUtilization`). Exit the `Test SNMP` dialog box.

Unfortunately, there aren't many options when a node does not support an SNMP variable used to compute health. You can do one of the following:

- Remove the Health Component altogether from the gauge's <*Summary*> entry in the `PortalView.xml` file. In this way, the associated SNMP variable/expression will not be used in computing health.
- Configure the `MgmtData` filter or the Portal View file's `NodeSelection` or `InterfaceSelection` such that only nodes supporting that MIB variable pass.

"Data unavailable" error message in one column of details table (for all nodes or interfaces)

Symptom:

"Data unavailable" in detail tables for all nodes.

Possible Cause A:

The node in question may not support one of the MIB variables used in computing that column value.

Solution:

See the Solution under Possible Cause D on page 144.

Possible Cause B:

There may be an error in the specification of the requested MIB variable or MIB expression, preventing NNM's `snmpCollect` process from performing any collections on this metric.

Solution:

Check for MIB variable/expression validity. See "Data unavailable" error message on one row of details table (for a particular node or interface) on the previous page.

For information about the steps required when the Service Information Portal software requests SNMP data from NNM, see *Presenting_NNM_Data.pdf*. Verify that each step of the process is working correctly.

Nodes or interfaces missing from details table

Symptom:

You expected more nodes to pass the filters than are displayed within the Detailed Network Health table.

Possible Cause A:

By default, only 20 rows are displayed within any Detailed Network Health table, although more than 20 may be included in the health calculation.

Solution:

By default, only the 20 least healthy nodes/interfaces are shown. To increase this value, increase the `maxDetail` attribute in the `netHealthConfig.xml` file. This setting affects all gauges within all defined portals.

If you increase the number of rows displayed and still have nodes missing, verify that the `MgmtData` filter, `NodeSelection` and/or `InterfaceSelection` filters are correctly defined.

Possible Cause B:

The missing nodes or interfaces may currently have status of "unknown" in the NNM object database. This happens when the device is unreachable from the NNM management station due to some connection device being down (such as a router).

Solution:

By default, devices with an "unknown" status are excluded from the details table. If you wish to include "unknown" devices, change the `showUnknown` attribute setting to "yes" in the `netHealthConfig.xml` file. This setting affects all gauges within all defined portals. The unknown devices, if any, will be placed at the bottom of the table, following any "known" devices.

Reading on the gauge does not match the values in the details table

Symptom:

The values displayed in the Detailed Network Health table do not seem to support the final value displayed on the gauge.

Possible Cause:

By default, only 20 rows are displayed within any Detailed Network Health table, although more than 20 may be included in the health calculation.

Solution:

By default, only the 20 least healthy nodes/interfaces are shown. To increase this value, increase the `maxDetail` attribute in the `netHealthConfig.xml` file. This setting affects all gauges within all defined portals.

Score for a node does not match the values given for its interfaces in the next lower level of details table

Symptom:

The values displayed for a node's interfaces in the Detailed Network Health table don't seem to support the value displayed for the overall node.

Possible Cause:

By default, only 20 rows are displayed within any Detailed Network Health table, although more than 20 may be included in the health calculation.

Solution:

By default, only the 20 least healthy nodes/interfaces are shown. To increase this value, increase the `maxDetail` attribute in the `netHealthConfig.xml` file. This setting affects all gauges within all defined portals.

Gauges are not available for me to add from the list of available Network Device Health Gauges

Symptom:

When displaying the list of available gauges, some are missing.

Possible Cause:

When you insert the Network Device Health module into a portal for the first time, all gauges defined within the `OVDefaultNetHealth.xml` file are copied into the *`PortalView`*`.xml` file. From that point on, only gauges physically defined within the current *`PortalView`*`.xml` file are displayed on the list.

Solution:

If a gauge that you wish to use is missing from the selection list, open a Portal View XML configuration file that contains the gauge and copy the gauge's definition (<*Summary*>) into the `PortalView.xml` file that you want to add it to.

What does the 100% health score mean? How do I display more information about how health scores are calculated?

Symptom:

I want to display more information about the health score calculation in the details table.

Possible Cause:

The `showRawData` attribute in the *`PortalView`*`.xml` file may be set to `NO`.

Solution:

To display the maximum amount of information about how health scores are calculated, set the `showRawData` attribute in the *`PortalView`*`.xml` file may be set to `YES`.

Note that `showRawData` applies to all *Summary* sections in that Network Device Health module instance.

The data collected seems to switch from one router interface to another

Symptom:

The data collected for a particular interface in a router is questionable.

Possible Cause:

Each time a router reboots, the SNMP interface index mapping is reconfigured. The ifIndex numbers assigned may drift from one interface to another. The NNM Data Collector is using `ifIndex` to identify interface instances. The collected data may be coming from a different interface after each router reboot.

Solution:

The drift of `ifIndex` numbers will stabilize when the Service Information Portal's data collection configuration is updated by executing the `ovcolautoconf` command on the NNM system. The problem only appears after router reboots.

Topology Map Module

"Data currently unavailable" message appears below a submap's title bar

Possible Cause A:

The NNM ovwdb process on an NNM management station is not running. The Topology module is dependent upon this process to supply information.

Symptom A: Log message: ERROR: Connection to OVW lost.

The portal_log file is located in the following directory:

Windows NT/2000: <SIP_install_dir>\log\sip.log

UNIX: /var/opt/OV/SIP/log/sip.log

Solution A:

Run the ovstart command on the NNM management station.

Possible Cause B:

ovw is not running on the server with the map open.

Symptom B:

Log message reads:
```
error Topology:Ovw    Thread-19    985636515095    An ovw
serving the map default was not found on the host
jorma.cnd.hp.com. Tried the following port(s):3700 3701
```

The sip.log file is located in the following directory:

Windows NT/2000: <SIP_install_dir>\log\sip.log

UNIX: /var/opt/OV/SIP/log/sip.log

Solution B:

Start ovw on the server with the map open.

Possible Cause C:

The map is running, but the map is running with a session number greater than 0 and there is a gap of *numMapRetries* in the sequence of

ovw session ports.

Symptom C:

See Symptom B.

Solution C:

Exit the ovw session and restart it so that it uses the lowest available session number.

Possible Cause D:

OVW authorization not configured on the remote server.

Symptom D:

Log message reads:
`Permission denied. The map default was not found on the host nganesan.cnd.hp.com`

The `sip.log` file is located in the following directory:

Windows NT/2000: `<SIP_install_dir>\log\sip.log`

UNIX: `/var/opt/OV/SIP/log/sip.log`

Solution D:

Modify `ovw.auth` and `ovwdb.auth` on remote server.

Possible Cause E:

`Wrong ovwDbPort` Specified in the `nmConfig.xml` file.
`Data Currently Unavailable`

Symptom E:

`error Topology:Ovw Thread-21 985710826476 Database not available`

Solution E:

Fix `ovwDBPort` setting in the `nmConfig.xml` file.

Possible Cause F:

Submap is not persistent.

Symptom F:

Log file reads: errorTopology: OvwThread-19988141835579 The submap name may be incorrect or if Customer Views is installed the submap name is not unique. If this is the case, specify the whole path.

Solution F:

Make the submap persistent.

Possible Cause G:

Submap name is misspelled, or if customer views is installed, the submap name may not be unique.

Symptom G:

See symptom F.

Solution G:

Specify the whole path. Correct spelling.

Possible Cause H:

Submap has been deleted and no longer exists

Symptom H:

See symptom F.

Solution H:

Remove submap name from configuration file or re-create submap.

Topology Map module hangs when trying to display a submap

Possible Cause A:

The NNM management station is in the pause state, for example for a backup procedure.

Solution A:

Wait until the backup complete (or run the `ovresume` command).

Possible Cause B:

There is an ovw running on the server that is hung. This may or may not be the ovw for the map having the submap to be displayed. (For example, a hung ovw can occur if you exit out of a Reflection X session without closing ovw.)

Solution B:

Check to make sure all ovw processes that are running are responding. If any of the ovw processes are hung, manually stop the process.

Possible Cause C:

It may not be hung but may just be taking a long time to find the map. This could occur if *numMapRetries* is high or there are many session number gaps between the running ovw sessions. Timeouts will generally only be a problem on Windows NT.

Solution C:

One solution is to exit and restart all the ovw sessions. This will restart the ovw sessions with contiguous session numbers.

Possible Cause D:

Another process on port 3600 or 3601. Check log file. Calling `OvwInitSession` on port.

Solution D:

Determine port configuration on the NNM management station.

"Managed objects not found" message is displayed in the submap area

Possible Cause A:

It may be that there are no symbols in the submap.

Possible Cause B:

It may be that a filter has been applied that results in no objects passing the filter for that particular submap.

Solution:

Change the filter that you are applying.

None of the icon symbols are displayed correctly

Possible Cause A:

No bitmaps on the local file system. If this is a first time installation, you have to get the bitmaps from the NNM station. Check if `symblRegSource` is set to "yes" (this is an attribute for the NNM station in `nmConfig.xml`).

Solution A:

Open `nmConfig.xml` and change `symbolRegSource` to "yes."

Possible Cause B:

Wrong `webSrvPort` specified in the `nmConfig.xml` file.

Symptom B:

No bitmaps found, only background shapes seen

```
error Topology:SymbolRefreshCache   Thread-21   985709926656
You might want to check the webSrvPort for
nganesan.cnd.hp.com:8880 java.net.ConnectException:
Connection refused: no further information
```

Solution B:

Fix port setting in `nmConfig.xml` file.

Some of the icon symbols are not displayed correctly

Possible Cause A:

Attribute in `nmConfig.xml` not set correctly.

Solution A:

In the `nmConfig.xml` file, the entry for this NNM management station might not have the `symbolRegSource` field set to "yes." See the `nmConfig.dtd` file for more information.

Background graphic for a submap is not displayed

Note: Only the default background graphics in the `<OpenView directory>/backgrounds` directory are certain to work across servers.

Possible Cause:

The background graphics file was not found in the expected location on the portal server.

Solution:

Install the background graphics on the portal server in the same location as the remote server.

"Currently not configured" message appears below Topology module title bar and no submap displays.

Possible Cause A:

Hostname in `PortalView.xml` file does not match host name in `nmconfig.xml`.

Currently not Configured
`error Topology:Ovw Thread-21 985710426012 The host nganesan.cnd.hp.com is not specified in nmConfig.xml`

Possible Cause B:

No stations in `nmConfig.xml`
Currently not Configured error NMConfig Thread-21 985710619802 There are no NNM stations configured in nmConfig.xml. At least one station must be configured for NNM modules to operate.

The Topology module opens slowly

Possible Cause:

You are getting map data from multiple NNM stations, and they all have the same applications installed (i.e., the same symbol information).

Solution:

Open the `nmConfig.xml` file on the SIP server. Improve the startup performance by specifying `symbolRegSource=yes` for just one NNM station and `symbolRegSource=no` for all the rest.

Glossary

A-B

authentication The process by which a user identifies and validates him/herself to the system.

authentication provider A configured component of the system that authenticates users that attempt to use the system.

authorization The granting of access privileges to an authenticated user that determines what the user can see and do while logged into the system.

C

configuration The combination of settings of software parameters and attributes that determine the way the software works, the way it is used, and/or how it appears.

configuration file A file that contains specifications or information that can be used for determining how a software program should look and operate.

configure To define and/or modify specified software settings to fulfill the requirements of a specified environment, application and/or usage.

current role The currently selected role for a logged in SIP user. The Role drop-down list box in the portal button bar shows the current role, and allows it to be changed.

customer data filtering The use of attributes as a mask for constraining the data that is to be acted on, used, or displayed in the user interface. In HP OpenView Service Information Portal, the information that is presented in the user interface for a given user login is filtered by the specific management data that is associated with a given role. Customer data filtering can also be described as “customer segmentation.”

customer model A mapping of customers to resources, where resources are associated hosts, interfaces, and services. A customer model can be defined in several XML files, or a mix of programs and files.

SIP 2.0 uses the so-called “Simple Customer Model” that is defined via an XML DTD.

customer model data source A configured URL or file that provides the mappings or partial mappings of customers to resources.

customize To design, construct and/or modify software to meet the needs and preferences of a particular customer or user. For HP OpenView Service

Information Portal, customizing is synonymous with assigning to customers what will be displayed to them. Customization tasks include customizing content, tabs, and tab layout, customer filtering, and the setting of options.

customization The process of designing, constructing and/or modifying software to meet the needs and preferences of a particular customer or user.

D

default role Any role in the User-Role Model that has the "defaultRole" attribute set to "yes." The default role is the role that is selected when a user logs into the portal and does not have a role configured for them. This is used only if there is no explicit user entry but there is a portal view file named for the login user. The default role is effectively a way to enable the "user-specific view file" mechanism for logging in. If no default role exists, there is no role to associate with the "user-specific view file" and the user is not authorized to use it. Only one role may be specified as the default role.

default user Any user in the User-Role Model that has the "defaultUser" attribute set to "yes". This user is selected when no user is found for a login after a user was authenticated.

E

edit permissions That which determines the editing operations that are available to a user through the program interface.

edit permissions level A group of operations that a user is authorized to perform through the program interface. Each level includes all the operations defined by the previous level and adds some additional operations.

extensible Software functionality whose capability, scope or effectiveness can be increased.

extend The act of increasing the capabilities, scope, and/or effectiveness of a program. The capabilities of HP OpenView Service Information Portal can be extended through the generic module and through the writing of XML.

F

filter A set of attributes and values that act as a pattern or mask through which data is passed. Filters allow matching-relevant information to be extracted and acted on while non-matching-irrelevant information is blocked.

L

login The string used to identify a user for authentication purposes.

M

Management Data Filter The security mechanism that defines what data is displayed through the portal.

message A communication using text and/or images. In HP OpenView Service Information Portal, messages are presented to customers via the message board module.

Message Board A module that is used for presenting messages to customers.

message content The information that is presented in a message. Message content may include text and/or graphics.

module A self-contained software component that performs a specific type of task or provides for the presentation of a specific type of data. Modules can interact with one another and with other software. In HP OpenView Service Information Portal, modules present specific sets of functionality to the user through the portal framework. Examples of modules include the Message Board, Service Browser, Network Device Health, and the Alarm Module.

module instance An instantiation of the module in a portal view file. Module Instance will likely differ from other instances in the portal view file by changing the XML description for that instance. For example, one could have an instance of the Alarms module displaying "All Alarms" and another instance displaying "Router Alarms."

P

page A single display or presentation of information on the World Wide Web. Typically a web page consists of an HTML file, referenced graphics files, and associated scripts.

portal A web site that provides a variety of different types of information and which serves as a gateway to other web sites. HP OpenView Service Information Portal consists of the framework and modules. It provides information and access to other websites through the modules and submodules.

portal framework A program that acts as the basic structure to support other software modules or programs that provide additional functionality for the user. In HP OpenView Service Information Portal, the framework provides a

mechanism for the modules to present information to the user. The framework also provides the structure for customization, configuration and extension of the portal's functionality.

portal view Consists of modules, tabs, and view properties. Each user account that is set up by an administrator has one or more roles, and each role is associated with one portal view. A portal view can be shared by multiple users.

portal view file A configuration file that contains specifications or information that can be used for determining how software should look to a given user. In HP OpenView Service Information Portal, portal view files are XML files that contain all information needed to render a portal view.

R

role That which defines what a user can see and do through the portal at a particular point in time. A role can be shared by multiple users.

role properties An extensibility mechanism used to provide authorization information associated with a role and that is not defined in the predefined role XML elements.

S

skin A setting that controls the visual appearance of the user interface. Skins can determine the color scheme, fonts, graphics, and other attributes presented in the user interface. The skins in the HP OpenView Service Information Portal are based upon W3C's Cascading Style Sheets. Existing skins may be extended, or new ones added to the 'css' files located under the `htdocs/styles` directory.

submodule A portion of a software module that provides a subset of the functionality provided by the module. A submodule performs a specific task or presents a specific set of data. In HP OpenView Service Information Portal submodules present different variations of the type of data presented by the Module. For example, one submodule of the Network Device Health Module presents Network health for Routers while another submodule presents Network health for Servers.

T

tab A page in the user interface that has a small index-card like projection. The projection typically presents the name for the page and allows navigation to the page by clicking. In HP OpenView Service Information Portal the main portal

pages have tabs. Service Information Portal provides one tab, but multiple tabs can be created using the Customize Content option on the Options page. Similar types of modules can be grouped together using tabs.

U

user preferences The attributes that are associated with a specific user. In Service Information Portal, user preferences control the name that appears in the portal header, and the color scheme, or “skins” that control the portal colors and fonts.

User-Role Model An authorization model that achieves security by associating users with roles and assigning to each role what the user is able to see and do. The User-Role Model consists of all User Role Package files.

Index

Index