Open NerveCenter^{TM} 3.8

Behavior Models Cookbook

UNIX and Windows

December 2002

Disclaimer

The information contained in this publication is subject to change without notice. OpenService, Inc. 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. OpenService, Inc. shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

Copyright

Printed in the USA, December 2002.

Open NerveCenter *Behavior Models Cookbook*

OpenService, Inc.
110 Turnpike Road, Suite 308
Westborough, MA 01581
Phone 978-366-0804
Fax 978-366-0814
http://www.open.com

Contents

Preface vii

Overview of this book viii

Overview of the Open NerveCenter Documentation and Help ix

Viewing the Online Documentation ix

Printing the Online Documentation xi

UNIX Man Pages for NerveCenter Commands xii

Getting Help While Working in NerveCenter xii

Accessing the Open NerveCenter Web Site xiii

Accessing the Open Technical Support Web Site xiii

Conventions xiii

Typographical Conventions xiii

Tips, Notes, and Cautions xiv

Chapter 1. Introduction 1

Using Behavior Models 1

Using Vendor-Specific Behavior Models 2

Chapter 2. Authentication Models 3

Authentication 4

Chapter 3. Data Collection Models 5

AllTraps_LogToFile 6

GatewayList 7

IfData_LogToFile 8

IPSweep (UNIX) 9

IPSweep (Windows) 10

NoSystemInfo 11

RoutingMon 12

Chapter 4. DBI Models 13

ifEntry_LogToDBI 14

Chapter 5. Interface Status Models 15

IfErrorStatus 16

IfLinkUpDown 18

IfLoad 20

IfUpDownStatus 22

IfUpDownStatusByType 23

Chapter 6. Node Status Models 29

NodeStatus 30

NodeStatusDwnStrm 32

Chapter 7. TCP Status Models 35

TcpConnMon 36

TcpRetransAlg 38

TcpRetransMon 40

Chapter 8. Troubleshooting Models 43

InformConnectStatus 44

Chapter 9. Vendor Models 45
CiscoBufferUtilization 47
CiscoCPUUtilization 49
CiscoCRCCheck 51
CiscoFanCheck 53
CiscoMemGetCheck 55
CiscoPacketsDrop 57
CiscoPwrSupplyCheck 59
CiscoTempCheck 61
CpqDaLogDrvStat 63
CpqNic 65
CpqDaPhyDrvThresh 66
CpqScsiPhyStat 67
CpqUnknown 69
CpqUps 70
SnmpresCritAppStatus 72
SnmpresDiskUsage 74
SnmpresGenericLogMonitor 76
SnmpresLoginMonitor 78
SnmpresMemUsage 80
SnmpresProcessorLoad 82
WfBufferUtilization 84
WfFanCheck 86
WfMemCheck 88
WfPwrSupplyCheck 90
WfTempCheck 92

Index 95

Preface

This preface will help you get the best results from the Open NerveCenter^{TM} documentation and help, which includes printed guides, online guides, help, and contact information for OpenService, Inc. and Open technical support.

This preface includes the following sections:

Section	Description
Overview of this book on page viii	Includes an overview of the contents of this guide and what you need to know before you use the guide.
Overview of the Open NerveCenter Documentation and Help on page ix	Lists and describes the components of the Open NerveCenter support system, including printed guides, online guides, help, and links to the Open NerveCenter Web site and the Open technical support Web site.
Conventions on page xiii	Describes the conventions that are used to help you find information quickly.

Overview of this book

The *Behavior Models Cookbook* describes each behavior model shipped with Open NerveCenter. Behavior models detect network conditions or events, correlate related conditions, and invoke appropriate actions in response. NerveCenter comes configured for immediate use, but you can easily customize it to meet the demands of your network environment. You can define groups of devices you want to monitor, specify the critical events or sequences of events you want to detect, and assign actions for NerveCenter to take.

Before reading this manual, you should be familiar with the following subjects and products: the Simple Network Management Protocol (SNMP), common MIB-II objects, your network management platform, and Perl scripting.

The *Behavior Models Cookbook* contains the following sections:

Title	Description
Chapter 1, *Introduction*	Describes how to use behavior models shipped with Open NerveCenter.
Chapter 2, *Authentication Models*	Discusses the network authentication behavior model.
Chapter 3, *Data Collection Models*	Describes various behavior models that perform information gathering tasks: list gateways, list devices missing system information, log information from SNMP traps, identify routing errors, log interface attribute values for ifEntry nodes, identify nodes using a ping sweep.
Chapter 4, *DBI Models*	Describes the behavior model that uses Perl to query and write to a database.
Chapter 5, *Interface Status Models*	Describes behavior models that monitors interface error rates, traffic, and status.
Chapter 6, *Node Status Models*	Describes the behavior models used for monitoring node status and for downstream alarm suppression.
Chapter 7, *TCP Status Models*	Discusses the behavior models used to monitor a network’s TCP connections and transmissions.
Chapter 8, *Troubleshooting Models*	Discusses the behavior model used to monitor the status of NerveCenter's connection with the OpenView Platform Adapter (OVPA) and the Inform Acknowledgement Queue.
Chapter 9, *Vendor Models*	Describes the groups of behavior models for use in monitoring Cisco and Wellfleet (Nortel) routers, and Compaq devices.

Overview of the Open NerveCenter Documentation and Help

Welcome to the Open NerveCenter support system—designed to give you the answers you need. This documentation explains important in-depth concepts, describes how to use NerveCenter, and provides answers to specific questions.

Refer to the following for more information:

- *Viewing the Online Documentation* on page ix
- *Printing the Online Documentation* on page xi
- *Getting Help While Working in NerveCenter* on page xii
- *Accessing the Open NerveCenter Web Site* on page xiii
- *Accessing the Open Technical Support Web Site* on page xiii

Viewing the Online Documentation

NerveCenter provides the complete set of books in HTML format. You must have a Web browser to view the HTML documents. You can use either Netscape Navigator or Microsoft Internet Explorer. Using other browsers can limit your ability to take advantage of documentation features such as full-text search. Refer to *Release Notes* for the browser versions supported with this release.

You can display the HTML documents:

- From the Start menu by selecting Programs, then OpenService NerveCenter, then Online Documentation (Windows)
- Opening the file `index.html` in a browser (UNIX). The `index.html` file is located in your *installation*`\docs` directory.
- By selecting Online Documentation from the Help menu in NerveCenter Client or NerveCenter Administrator.

When you have the online documentation displayed in your browser, select the document you want to view from the tree in the left pane.

To facilitate navigation within the documentation files, each topic provides the name and a hypertext link to its previous and next topic, as well as to the book, chapter, and heading in which the topic is located. This information is displayed at the top of each page.

If you are not sure what book you need, use the full-text search feature to enter one or more terms and then search for topics that contain the text you entered.

Tip Instructions on using the online documentation and search feature can be accessed from the How to use this documentation link in the browser’s window.

The following table lists the documents that ship with NerveCenter..

Table 3-1. The NerveCenter Library

Book Title	Description	Application	Audience	PDF for Print
Release Notes	Describes the features that are new in NerveCenter v3.8 and includes up-to-date information, software support, corrections, and instructions.	All	All	`relnotes.pdf`
Getting a Quick Start with NerveCenter	Provides a preview prior to installing and configuring NerveCenter for your network. Takes approximately one to two hours to complete.	All	New users	`qckstart.pdf`
Upgrading to NerveCenter 3.8	Explains how to upgrade your current version of NerveCenter to NerveCenter 3.6.	All	Installation team	`upgrading.pdf`
Installing NerveCenter	Helps you plan and carry out your NerveCenter installation. Use the *Release Notes* in conjunction with this book.	All	Installation team	`install.pdf`
Managing NerveCenter	Explains how to customize and tune NerveCenter after it has been installed.	NerveCenter Administrator	Administrator	`managing.pdf`
Integrating NerveCenter with a Network Management Platform	Explains how to integrate NerveCenter with one or more supported network management platforms.	NerveCenter Administrator	Administrator	`integrating.pdf`
Learning How to Create Behavior Models	Provides step-by-step instructions with examples for creating new behavior models.	NerveCenter Client	Users with administrative privileges	`learning.pdf`
Designing and Managing Behavior Models	Explains behavior models in depth, how to create new models or modify existing ones, and how to manage your models.	NerveCenter Client	Users with administrative privileges	`design.pdf`
Monitoring Your Network	Explains how NerveCenter works and how you can monitor your network most effectively.	NerveCenter Client and Web Client	Users	`monitor.pdf`
Behavior Models Cookbook	Describes each behavior model shipped with Open NerveCenter.	NerveCenter Client	Users with administrative privileges	`cookbook.pdf`
Open NerveCenter: Downstream Alarm Suppression	White paper that explains how NerveCenter handles downstream alarm suppression.	NerveCenter Client	Users with administrative privileges	`downstream.pdf`

Table 3-1. The NerveCenter Library (continued)

Book Title	Description	Application	Audience	PDF for Print
NerveCenter 3.8: An Overview	White paper that describes the components of NerveCenter and explains how these components interact with each other. You will also learn how NerveCenter interacts with external software, such as a network management platform.	NerveCenter Client and Administrator	Administrator and users with administrative privileges	`NCOverview.pdf`
NerveCenter Integration with Micromuse Netcool/OMNIbus	White paper that describes how NerveCenter works with Micromuse Netcool/OMNIbus to help enterprises monitor the status of their networks.	NerveCenter Client and Administrator	Administrator and users with administrative privileges	`netcoolIntegrate.pdf`
NerveCenter: Node Classification	White paper that describes how NerveCenter classifies the SNMP version of a node.	NerveCenter Client	Administrator and users with administrative privileges	`nodeclass.pdf`
Quick reference cards	PDF only. NerveCenter quick reference cards provide convenient reference material for some of the more common NerveCenter tasks. The following cards are provided: ◆ Reference for monitoring NerveCenter. ◆ Reference for installing NerveCenter. ◆ Reference for using behavior models.	NerveCenter Client and Administrator	All	`quickreference.pdf`

Printing the Online Documentation

The NerveCenter books, papers, and quick reference cards in the library are also available as Portable Document Format (PDF) files that you can open and print.

Note You must have Adobe Acrobat Reader installed to open or print the PDF files. You can download Reader free of charge from Adobe’s Web Site at the following address: http://www.adobe.com

All PDF files are located in your *installation*`\docs` directory. The three quick reference cards are consolidated in one file named quickreference.pdf.

In addition to the files in the *installation*\docs directory, four PDF files are also located in the root directory of your CD so you can print them prior to installation. These four books are: *Getting a Quick Start with NerveCenter* (qckstart.pdf), *Installing NerveCenter* (install.pdf), *Release Notes* (relnotes.pdf), and *Upgrading to NerveCenter 3.8* (upgrade.pdf).

UNIX Man Pages for NerveCenter Commands

On UNIX systems, you can use NerveCenter man pages to find reference and usage information about NerveCenter commands that you view from the UNIX shell as you would any other system man page.

When you specify documentation during NerveCenter installation, the script installs nroff-tagged man pages and automatically updates your system's MANPATH environment variable to point to the NerveCenter man page directory.

Getting Help While Working in NerveCenter

On UNIX and Windows, use NerveCenter online help to get immediate answers to your questions about particular fields and windows. You can access help topics from the Help menu in any NerveCenter application. In addition, context-sensitive help is available for NerveCenter windows and dialog boxes. If you are using Windows, you can also obtain context-sensitive help for data-entry fields.

The following table describes the ways context-sensitive help can be invoked:

Table 3-2. Methods for Getting Help

Platform	Screen-level Help	Field-level Help
Windows	Select the Help button in any window to display general information about the window.	Select a data-entry field and then press F1 or Shift+F1 to display information about the field.
UNIX	Press F1 or select the Help button in any window to display general information about the window.	No context-sensitive field-level help is available. Descriptions of data-entry fields are available in the help topics.

To view NerveCenter help files on UNIX, we recommend that you set your system fonts to medium or large. If the help topics still do not display correctly, choose Large Font from the Options menu in the help window.

Accessing the Open NerveCenter Web Site

Use the Open NerveCenter Web site for current product information, downloads, and information about product support. The address for this Web site is:

- http://www.open.com

Accessing the Open Technical Support Web Site

Use the Open Tech Support Web site to read current white papers related to NerveCenter, search the knowledge base for answers to technical support questions, view the NerveCenter Frequently Asked Questions (FAQ) page, automatically receive notice of product updates, post questions to the Open support staff, contact Technical Support, find out about NerveCenter training, and get the latest patches, upgrades, and utilities. The address for this Web site is

- http://www.open.com

Conventions

The screen shots in this book were taken on Windows. Unless otherwise noted, the application is the same on Windows and UNIX. For cases where the application is different on UNIX, a UNIX screen shot has been included as well.

You should use the appropriate conventions for your platform. For example, when specifying a path, use backslashes on Windows and slashes on UNIX. Significant differences between the platforms are noted in the text.

Typographical Conventions

The typographical conventions used throughout the support system are described in the following table:

Table 4. Conventions

Convention	Description
GUIFont	Used to depict graphical user interface (GUI) objects, such as fields, list boxes, menu commands, and so on. For example: Enter your password in the Password field.
Italics	Used for placeholder text, book titles, new terms, or emphasis. Replace placeholder text with your specific text. For example, replace *filename* with the name of your file. Do *not* use file names that contain spaces.
`Code`	Used to show what commands you need to type, to identify pathnames where files are located, and to distinguish system or application text that is displayed to you or that is part of a code example.

Table 4. Conventions (continued)

Convention	Description
Key+Key	Used to show that you must hold down the first key while pressing the second key. For example, Ctrl+S means hold down the Ctrl key while you press S.
Toolbar buttons	If toolbar buttons are available, they are displayed in the margin next to the step in which you can use them. Other shortcuts are noted as tips. (Shortcut, or accelerator, keys are displayed next to their respective menu choices in the applications themselves.)

Tips, Notes, and Cautions

Tips, notes, and cautions are used to emphasize information. The following samples describe when each is used.

Tip Used for nice-to-know information, like a shortcut.

Note Used for important information that you should know, but that shouldn’t cause any damage to your data or your system if you choose to ignore it.

Caution Used for information that will prevent a problem. Ignore a caution at your own risk.

Introduction 1

This chapter describes how to use the behavior model shipped with NerveCenter^{TM}.

Using Behavior Models

If this is a vendor-specific behavior model (Cisco, Compaq, or Wellfleet (Nortel)), go to the section *Using Vendor-Specific Behavior Models* Otherwise, to use this behavior model, perform these steps:

❖ How to use a predefined behavior model:

1. Import the model. (See the book *Designing and Managing Behavior Models*.)
2. (Optional) Choose to customize certain settings for the model, such as poll and timer intervals, counters, thresholds, alarm actions, and so on. See the section *Optional customization* in the description of each behavior model for suggestions on customizing a behavior model. For details about how to customize various elements of the behavior model, see *Designing and Managing Behavior Models*.
3. Turn on the model.
   a. From the NerveCenter Client's Admin menu, choose Alarm Definition List.
   b. Right-click the model's alarm(s) and choose On.

Using Vendor-Specific Behavior Models

To use this vendor-specific behavior model (Cisco, Compaq, or Wellfleet (Nortel)), perform these steps:

❖ How to use a vendor-specific behavior model:

1. Decide which vendor’s base objects you want to monitor and note each object’s `.asn` file under:

UNIX	Windows
`mibs/vendors/`*vendor-name*	`Mib\vendors\`*vendor-name*

2. Add or remove the necessary MIB definitions in:

UNIX	Windows
`mibs/mibcomp.txt`	`Mib\mibcomp.txt`

   Refer to the book, *Managing NerveCenter*.

3. Compile the NerveCenter MIB. (Refer to the book, *Managing NerveCenter*.)

4. Reload the NerveCenter MIB. (Refer to the book, *Managing NerveCenter*.)

Follow the steps in the preceding section,*Using Behavior Models* on page 1.

Authentication Models 2

The authentication models reside under `models/authentication`, and consist of:

Section	Description
Authentication on page 4	Tracks authentication failures on the network.

Authentication

Path: `/model/authentication/authentication.mod`

This behavior model tracks authentication failures on the network. If more than three failures occur within ten minutes, then NerveCenter sends a 1101 Inform to the network management platform.

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- AuthFail (poll)
- AuthQuickFail (poll)
- AuthFail (mask)
- Authentication (alarm)

Events Sent to OpenView. Table 2-1 summarizes the events this behavior model sends to OpenView:

Table 2-1. Events Sent to OpenView by Authentication

Event Name	Inform Number	Description
NC_Authentication	1101	This inform event is triggered by the Authentication alarm and indicates that three consecutive log in attempts failed for the node. This event could indicate that unauthorized access to the network has been attempted.

For more information about integrating NerveCenter with OpenView or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Change Inform to Inform Platform for Netcool, TME, Unicenter

Data Collection Models 3

The data collection models reside under `models/data_collection`, and consist of:

Section	Description
AllTraps_LogToFile on page 6	Repeatedly logs the information contained in all SNMP traps detected by NerveCenter.
GatewayList on page 7	Identifies gateways and outputs them in the NerveCenter Client's Alarm Summary window.
IfData_LogToFile on page 8	Repeatedly logs noteworthy interface attribute values for every managed, non-suppressed node that contains the ifEntry property.
IPSweep (UNIX) on page 9	(UNIX version) Identifies responsive nodes in the network that are not in the Open NerveCenter database.
IPSweep (Windows) on page 10	(Windows version) Identifies responsive nodes in the network that are not in the Open NerveCenter database.
NoSystemInfo on page 11	Determines which devices are missing system information (name, description, and contact).
RoutingMon on page 12	Monitors routing activity by tracking nodes that cross specified thresholds for: ◆ ICMP Destination Unreachable messages ◆ Time exceeded messages ◆ IP datagrams discarded because no route could be found to deliver them

AllTraps_LogToFile

Path: `/model/data_collection/alltraps_logtofile.mod`

This Behavior model repeatedly logs the information contained in all SNMP traps detected by NerveCenter. Each alarm instance represents a single node.

On each transition, NerveCenter performs a “Log To File” alarm action so that reports on SNMP traps can be produced.

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- AllTraps (mask)

Events Sent to OpenView. AllTraps_LogToFile sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)

GatewayList

Path: `/model/data_collection/gatewaylist.mod`

This behavior model identifies gateways—devices that forward IP packets—in a managed network. NerveCenter outputs all identified gateways in the NerveCenter Client’s Alarm Summary window.

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- IfGateway (poll)

Events Sent to OpenView. GatewayList sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Set poll conditions to detect additional information, such as gateways with high traffic, gateways with errors, and so on.
- Add a NerveCenter alarm action to write the list of gateways to a log or to an xterm window.

IfData_LogToFile

Path: `/model/data_collection/ifdata_logtofile.mod`

This Behavior model repeatedly logs noteworthy interface attribute values for every managed, non-suppressed node that contains the ifEntry property.

On each transition, NerveCenter performs a “Log To File” alarm action so that reports on interface attribute values can be produced.

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- IfData (poll)

Events Sent to OpenView. IfData_LogToFile sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Change ifEntry attributes polled to suit your own network management strategy.
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)

IPSweep (UNIX)

Path: `/model/data_collection/ipsweep_unix.mod`

(UNIX version) This behavior model identifies responsive nodes in the network that are not in the Open NerveCenter database. When NerveCenter finds such a node, it sends an SNMP trap to either the node source (if one is specified in NerveCenter) or the local host, so that `ipsweep` can either prompt population of the node database in the network management platform or in Open NerveCenter.

IPSweep runs the ipsweep executable whenever the alarm transitions to the PingSweep state. By default, the path for `ipsweep` is:

```
/opt/OSInc/bin
```

If `ipsweep` does not reside on this default path, you will have to modify the two Command alarm action instances in the alarm definition for IPSweep with the correct path. For information on the Command alarm action, refer to *Designing and Managing Behavior Models*.

Note If you are using the IPSweep alarm to populate a network management platform's database, turn off your network management platform's network topology discovery.

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- TruePoll (poll)
- IPSweep (alarm)

Events Sent to OpenView. IPSweep sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set the timer to perform the ping sweep
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)

IPSweep (Windows)

Path: `\model\data_collection\ipsweep_nt.mod`

(Windows version) This behavior model identifies responsive nodes in the network that are not in the Open NerveCenter database. When NerveCenter finds such a node, it sends an SNMP trap to either the node source (if one is specified in NerveCenter) or to the local host, so that `IPSweep.EXE` can either prompt population of the node database in the network management platform or in Open NerveCenter.

IPSweep runs `IPSweep.exe` whenever the alarm transitions to the PingSweep state. By default, the path for `IPSweep.exe` is:

```
C:\program files\OpenService\nervecenter\bin
```

If `IPSweep.exe` does not reside on this default path, you will have to modify the two Command alarm action instances in the alarm definition for IPSweep with the correct path. For information on the Command alarm action, refer to *Designing and Managing Behavior Models*.

Note If you are using the IPSweep alarm to populate a network management platform’s database, turn off your network management platform’s network topology discovery.

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- TruePoll (poll)
- IPSweep (alarm)

Events Sent to OpenView. IPSweep sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set the timer to perform the ping sweep
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)

NoSystemInfo

Path: `/model/data_collection/nosysteminfo.mod`

This behavior model determines which devices are missing system information (name, description, and contact). If you assign trouble tickets based on the value in the contact field, for example, that information is very important.

You can use this model as a template to create other alarms that verify the presence of particular MIB data that you require in your environment.

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- GotSystemInfo (poll)

Events Sent to OpenView. NoSystemInfo sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- If a subset of managed nodes is required to have system data initialized, change the GotSystemInfo poll's property to target those devices.
- Add actions to the noSystemInfo transition to log the occurrence or send e-mail to the person responsible for maintaining the agent's system data. Add an SNMP Set action to set the system data if the attribute values are known and the agents are configured to allow writes.

RoutingMon

Path: `/model/data_collection/routingmon.mod`

This behavior model monitors routing activity by tracking nodes that cross specified thresholds for:

- ICMP Destination Unreachable messages
- Time exceeded messages
- IP datagrams discarded because no route could be found to deliver them

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- RoutingDestUnreach (poll)
- RoutingNoRoutes (poll)
- RoutingTTLExpired (poll)

Events Sent to OpenView. RoutingMon sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Poll conditions for ICMP message and IP datagram thresholds

DBI Models 4

The DBI models reside under `models/DBI`, and consist of:

Section	Description
ifEntry_LogToDBI on page 14	Collects ifEntry metrics and logs them to a database.

ifEntry_LogToDBI

Path: `/model/DBI/ifEntry_LogToDBI.mod`

Collects ifEntry metrics and logs them to a database.

The model takes advantage of NerveCenter's single threaded Perl interpreter by establishing a connection and allowing each successive use (whether the same node or not) to utilize the connection. Only in the event when a connection is dropped is an attempt made to re-establish.

The ifEntry data is collected via normal polls and then inserted into the database.

The table schema is as follows:

Table 4-1. Schema for IfEntry_LogToDBI

Field	Type	Null	Key	Default	Extra
ipAddress	varchar(16)	YES		NULL	
ifIndex	int(11)	YES		NULL	
ifInOctets	int(11)	YES		NULL	
ifOutOctets	int(11)	YES		NULL	
ifSpeed	int(11)	YES		NULL	
timestamp	int(11)	YES		NULL	

Note The ifEntry_LogToDBI alarm works only with mySQL database.

Property. The following property must be in the property group for the nodes you want to monitor:

ifEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- DBI_OK (poll)
- DBI_NoConnect (poll)

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Update username and password information
- Add/Delete ifEntry metrics (and update database table)
- Poll intervals
- Update DSN script to utilize a different database

Interface Status Models 5

The interface status models reside under `models/interface_status`, and consist of:

Section	Description
IfErrorStatus on page 16	Monitors error rates on interfaces.
IfLinkUpDown on page 18	Monitors node interfaces.
IfLoad on page 20	Determines whether device load is low, medium, or high.
IfUpDownStatus on page 22	Determines whether each interface on a managed node is up, down, or testing.
IfUpDownStatusByType on page 23	Manages the following types of interfaces: ◆ Asynchronous Transfer Mode (ATM) ◆ Integrated Services Digital Network (ISDN) ◆ Fiber Distributed Data Interface (FDDI) ◆ Frame Relay Permanent Virtual Circuit (PVC) subinterfaces ◆ Frame Relay ◆ Local Area Network (LAN) ◆ Switched Multimegabit Data Service (SMDS) ◆ Synchronous Optical Network (SONET) ◆ Wide Area Network (WAN)

IfErrorStatus

Path: `/model/interface_status/iferrorstatus.mod`

This multi-alarm behavior model monitors error rates on interfaces. When the interface error rates are high, NerveCenter sends a 7000 Inform to the platform. If a high error rate persists, NerveCenter sends a 7001 Inform to the platform. You can trigger other alarms that use vendor-specific MIBs to further refine the problem and its cause.

Two additional alarms, IfErrorStatus_LogToDB and IfErrorStatus_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on interface error rates can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

ifEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- IfErrorRates (poll)

Events Sent to OpenView. Table 5-1 summarizes the events this behavior model sends to OpenView:

Table 5-1. Events Sent to OpenView by IfErrorStatus

Event Name	Inform Number	Description
NC_Interface_Error	7000	The error rate for this interface is high.
NC_Interface_ErrPers	7001	The error rate for this interface has been high over a persistent period of time.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set counters for persistence
- Poll intervals
- Poll conditions for error rates

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

IfLinkUpDown

Path: `/model/interface_status/iflinkupdown.mod`

This multi-alarm behavior model monitors node interfaces. When an interface goes down and remains down for a specified time (the default is three minutes), NerveCenter sends a 7004 Inform to the platform. When an interface comes back up, the alarms return to ground. The monitored nodes must be associated with a property group that contains the property ifEntry.

Two additional alarms, IfLinkUpDown_LogToDB and IfLinkUpDown_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on interface up/down status can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

ifEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- LinkDown (mask)
- LinkUp (mask)
- IfLinkUpDown (alarm)
- IfLinkUpDown_LogToDB (alarm)
- IfLinkUpDown_LogToFile (alarm)

Events Sent to OpenView. Table 5-2 summarizes the events this behavior model sends to OpenView:

Table 5-2. Events Sent to OpenView by IfLinkUpDown

Event Name	Inform Number	Description
NC_Link_Down	7004	Linkdown trap received from interface.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

IfLoad

Path: `/model/interface_status/ifload.mod`

Determines whether device load is low, medium, or high. Load is the amount of interface traffic compared to the media's capacity, expressed as a percentage. There are two definitions for each—one for point-to-point networks and one for broadcast networks.

This model can give an immediate impression of network utilization; however, it can also be used to track trends based on time, day, or location; to identify runaway processes that are using network resources; to help plan for network reconfiguration; and to help in any task that requires knowledge of traffic levels.

Two additional alarms, IfLoad_LogToDB and IfLoad_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on device load levels can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

ifEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- IfLoadRates (poll)

Events Sent to OpenView. Table 5-3 summarizes the events this behavior model sends to OpenView:

Table 5-3. Events Sent to OpenView by IfLoad

Event Name	Inform Number	Description
NC_Interface_Load	7003	The throughput for this interface has been high over a persistent period of time.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set counters for persistence
- Poll conditions for device load thresholds

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

IfUpDownStatus

Path: `/model/interface_status/updown/ifupdownstatus.mod`

This multi-alarm behavior model determines whether each interface on a managed node is up, down, or testing. If an interface is down, NerveCenter sends a 7002 Inform to the network management platform.

Two additional alarms, IfUpDownStatus_LogToDB and IfUpDownStatus_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on interface status can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

ifEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- IfStatus (poll)

Events Sent to OpenView. Table 5-4 summarizes the events this behavior model sends to OpenView:

Table 5-4. Events Sent to OpenView by IfUpDownStatus

Event Name	Inform Number	Description
NC_Interface_Down	7002	It has been determined that the interface is operationally down.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

IfUpDownStatusByType

Path: `/model/interface_status/updown_bytype/ifupdownstatus_bytype.mod`

IfUpDownStatusByType is one of the multi-alarm behavior models shipped with NerveCenter and provides interface management for devices that can be managed using the MIB-II and Frame Relay MIBs. This model manages the following types of interfaces:

- Asynchronous Transfer Mode (ATM)
- Integrated Services Digital Network (ISDN)
- Fiber Distributed Data Interface (FDDI)
- Frame Relay Permanent Virtual Circuit (PVC) subinterfaces
- Frame Relay
- Local Area Network (LAN)
- Switched Multimegabit Data Service (SMDS)
- Synchronous Optical Network (SONET)
- Wide Area Network (WAN)

The majority of the alarms in this model are subobject scope alarms that categorize an interface (the possible categories are listed above) and then monitor its status. For most interfaces, the interface can be up, down, or in testing mode. (The exception is a Frame Relay PVC, which can only be up or down.)

When an alarm instance transitions to one of these states, it executes an Inform action to notify OpenView Network Node Manager of the new state. For this Inform action to have the desired effect, you must integrate the `trapd.conf.txt` file supplied with these models with the standard NerveCenter `trapd.conf`.

The interface status alarms are listed below:

- IF-IfATMStatus
- IF-IfFDDIStatus
- IF-IfFramePVCStatus
- IF-IfFrameRelayStatus
- IF-IfISDNStatus
- IF-IfLANStatus
- IF-IfSMDSStatus
- IF-IfSonetStatus
- IF-IfWANStatus

The model file also includes three other alarms: IF-IfStatus, IF-IfColdWarmStart, and IF-IfNmDemand.

Property. The following property must be in the property group for the nodes you want to monitor:

frCircuitEntry (IF-IfFramePVCStatus) and ifEntry (all other alarms)

Trigger Generators. Below is a list of the objects that generate triggers:

- If-ColdWarmStart (poll)
- If-IfAdminOp (poll)
- If-IfDown (poll)
- If-IfDown2 (poll)
- If-IfFramePVC (poll)
- If-IfFramePVCDown (poll)
- If-IfFramePVCUp (poll)
- If-IfNotAdminOp (poll)
- If-IfTesting (poll)
- If-IfTesting2 (poll)
- If-IfTypeNotAdminOp (poll)
- If-ColdStart (mask)
- If-FramePVCUpDown (mask)
- If-LinkDown (mask)
- If-LinkUp (mask)
- If-WarmStart (mask)
- IF-IfATMStatus (alarm)
- IF-IfColdWarmStart (alarm)
- IF-IfFDDIStatus (alarm)
- IF-IfFramePVCStatus (alarm)
- IF-IfFrameRelayStatus (alarm)
- IF-IfISDNStatus (alarm)
- IF-IfLANStatus (alarm)
- IF-IfNmDemand (alarm)
- IF-IfSMDSStatus (alarm)

- IF-IfSonetStatus (alarm)
- IF-IfStatus (alarm)
- If-IfWANStatus (alarm)
- IF-SelectType (Perl subroutine)

Events Sent to OpenView. Table 5-5 summarizes the events this behavior model sends to OpenView:

Table 5-5. Events Sent to OpenView by IfUpDownStatusByType

Event Name	Inform Number	Description
NC_Interface_ColdWarmStart	1500	As a result of a device reboot, interface instances are no longer guaranteed to be valid. Thus, interface management has been restarted for this device.
NC_Interface_IfLanUp	1501	A LAN interface on this device has returned to an operational state.
NC_Interface_IfLanTesting	1502	A LAN interface desired to be operational on this device is in a testing state. No packets will be passed.
NC_Interface_IfLanDown	1503	A LAN interface desired to be operational on this device is in a down state.
NC_Interface_IfWanUp	1504	A WAN interface on this device has returned to an operational state.
EVENT NC_Interface_IfWanTesting	1505	A WAN interface desired to be operational on this device is in a testing state. No packets will be passed.
NC_Interface_IfWanDown	1506	A WAN interface desired to be operational on this device is in a down state.
NC_Interface_IfFrameRelayUp	1507	A Frame Relay interface on this device has returned to an operational state.
NC_Interface_IfFrameRelayTesting	1508	A Frame Relay interface desired to be operational on this device is in a testing state. No packets will be passed.
NC_Interface_IfFrameRelayDown	1509	A Frame Relay interface desired to be operational on this device is in a down state.
NC_Interface_IfFrameRelayPVCUp	1510	A Frame Relay PVC on this device has returned to an operational state.
NC_Interface_IfFrameRelayPVCDown	1511	A Frame Relay PVC desired to be operational on this device is in a down state.

Table 5-5. Events Sent to OpenView by IfUpDownStatusByType (continued)

Event Name	Inform Number	Description
NC_Interface_IfATMUp	1512	An ATM interface on this device has returned to an operational state.
NC_Interface_IfATMTesting	1513	An ATM interface desired to be operational on this device is in a testing state. No packets will be passed.
NC_Interface_IfATMDown	1514	An ATM interface desired to be operational on this device is in a down state.
NC_Interface_IfFDDIUp	1515	A FDDI interface on this device has returned to an operational state.
NC_Interface_IfFDDITesting	1516	A FDDI interface desired to be operational on this device is in a testing state. No packets will be passed.
NC_Interface_IfFDDIDown	1517	A FDDI interface desired to be operational on this device is in a down state.
NC_Interface_IfSONETUp	1518	A SONET interface on this device has returned to an operational state.
NC_Interface_IfSONETTesting	1519	A SONET interface desired to be operational on this device is in a testing state. No packets will be passed.
NC_Interface_IfSONETDown	1520	A SONET interface desired to be operational on this device is in a down state.
NC_Interface_IfISDNUp	1521	A ISDN interface on this device has returned to an operational state.
NC_Interface_IfISDNTesting	1522	A ISDN interface desired to be operational on this device is in a testing state. No packets will be passed.
NC_Interface_IfISDNDown	1523	A ISDN interface desired to be operational on this device is in a down state.
NC_Interface_IfSMDSUp	1524	A SMDS interface on this device has returned to an operational state.
NC_Interface_IfSMDSTesting	1525	A SMDS interface desired to be operational on this device is in a testing state. No packets will be passed.
NC_Interface_IfSMDSDown	1526	A SMDS interface desired to be operational on this device is in a down state.

Table 5-5. Events Sent to OpenView by IfUpDownStatusByType (continued)

Event Name	Inform Number	Description
NC_IntIfNnmDemand	1527	An Open NerveCenter initiated event which causes NNM to perform a nmdemandpoll on a particular node in order to force NNM to poll the node and correctly reflect the status of its interfaces.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)

Node Status Models 6

The node status models reside under `models/node_status`, and consist of:

Section	Description
NodeStatus on page 30	Monitors the status of nodes and, optionally, their SNMP agents.
NodeStatusDwnStrm on page 32	A downstream alarm suppression behavior model that uses node relationships to accurately monitor their status in a complex network.

NodeStatus

Path: `/model/node_status/nodestatus_models.mod`

This multi-alarm behavior model monitors the status of nodes and, optionally, their SNMP agents.

The SnmpStatus alarm monitors the status of nodes and their SNMP agents. Uses built-in triggers to determine if a device is up, down, or unreachable. Monitors nodes associated with a property group that contains the property system.

The IcmpStatus alarm monitors the status of nodes based on their response to ICMP echo requests (pings). Determines if a device is up, down, or unreachable. Nodes must be associated with a property group that contains the property icmpStatus, but they are not required to run SNMP agents.

Property. The following property must be in the property group for the nodes you want to monitor:

system (SnmpStatus alarm) and icmpStatus (IcmpStatus alarm)

Trigger Generators. Below is a list of the objects that generate triggers:

- IS_IcmpFastPoll (poll)
- IS_IcmpPoll (poll)
- SnmpPoll (poll)
- SnmpFastPoll (poll)
- SS_IcmpFastPoll (poll)
- SS_IcmpPoll (poll)
- ColdStart (mask)
- WarmStart (mask)
- SS_IcmpError (Perl subroutine)

Events Sent to OpenView. Table 6-1 summarizes the events this behavior model sends to OpenView:

Table 6-1. Events Sent to OpenView by NodeStatus

Event Name	Inform Number	Description
NC_IcmpTimeout	1102	This inform event indicates that attempts to contact a node have failed due to ICMP timeout.
NC_SnmpStatus	1103	This inform event is triggered by the SNMP timeout and indicates that attempts to contact a node have failed.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Node suppression on/off
- Change Inform to Inform Platform for Netcool, TME, Unicenter

NodeStatusDwnStrm

Path: `/model/node_status/nodestatus_dwnstrm.mod`

NodeStatusDwnStrm is a downstream alarm suppression behavior model that monitors nodes in a complex network. Using topology information—either from HP OpenView, or a file that you provide—this model uses the relationships between nodes to determine the status of those nodes accurately.

A multi-alarm behavior model, NodeStatusDwnStrm consists of the following alarms:

- **DwnStrmSnmpStatus**—accurately monitors the status of nodes that have an SNMP agent by taking into consideration the status of the nodes' parents.
- **DwnStrmIcmpStatus**—accurately monitors the status of nodes that do not have an SNMP agent by taking into consideration the status of the nodes' parents.
- **DwnStrmSnmpStatus_LogToDB** and **DwnStrmIcmpStatus_LogToDB**—perform a Log To Database alarm action on each transition so that reports on node status can be produced.

Note The LogToDB alarms, work on Windows only.

For more information about NodeStatusDwnStrm, refer to the white paper, *Open NerveCenter: Downstream Alarm Suppression*.

Property. The following property must be in the property group for the nodes you want to monitor:

system (SnmpStatus alarm) and icmpStatus (IcmpStatus alarm)

Trigger Generators. Below is a list of the objects that generate triggers:

- IS_IcmpFastPoll (poll)
- IS_IcmpPoll (poll)
- SnmpPoll (poll)
- SnmpFastPoll (poll)
- SS_IcmpFastPoll (poll)
- SS_IcmpPoll (poll)
- ColdStart (mask)
- WarmStart (mask)
- SetNodeStatusDown (Perl subroutine)
- SetNodeStatusTesting (Perl subroutine)
- SetNodeStatusUnreachable (Perl subroutine)
- SetNodeStatusUp (Perl subroutine)

- SS_IcmpError (Perl subroutine)
- TestParentSetNode (Perl subroutine)
- TestParentStatus (Perl subroutine)

Events Sent to OpenView. The table below summarizes the events this behavior model sends to OpenView:

Table 6-2. Events Sent to OpenView by NodeStatusDwnStrm

Event Name	Inform Number	Description
NC_IcmpTimeout	1102	This inform event indicates that attempts to contact a node have failed due to ICMP timeout.
NC_SnmpStatus	1103	This inform event is triggered by the SNMP timeout and indicates that attempts to contact a node have failed.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Node suppression on/off
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To Database alarm action options
  (log identity number and log action variables)

TCP Status Models 7

The TCP status models reside under `models/tcp_status`, and consist of:

Section	Description
TcpConnMon on page 36	Monitors the number of current TCP connections on each device and compares it with the maximum number.
TcpRetransAlg on page 38	Checks all devices that support the tcp MIB-II group and verifies that their transmission algorithm is the most efficient.
TcpRetransMon on page 40	Monitors the percentage of TCP retransmissions for a particular device.

TcpConnMon

Path: `/model/tcp_status/tcp_conn_mon.mod`

Monitors the number of current TCP connections on each device and compares it with the maximum number. Ignores devices that have dynamically-allocated maximum connections. Gives early warnings by detecting increasing levels of TCP connection saturation. When a device reaches 98 percent of the maximum TCP connections, NerveCenter sends a 7005 Inform to the platform.

Two additional alarms, TcpConnMon_LogToDB and TcpConnMon_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on TCP connections can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

tcp

Trigger Generators. Below is a list of the objects that generate triggers:

- TcpConnectionMon (poll)

Events Sent to OpenView. Table 7-1 summarizes the events this behavior model sends to OpenView:

Table 7-1. Events Sent to OpenView by TcpConnMon

Event Name	Inform Number	Description
NC_Tcp_Conn	7005	The current number of TCP connections is reaching a saturated state.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/tcp_status
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Poll conditions for TCP connection thresholds

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

TcpRetransAlg

Path: `/model/tcp_status/tcp_retrans_alg.mod`

Checks all devices that support the tcp MIB-II group and verifies that their transmission algorithm is the most efficient. An inefficient retransmission algorithm can severely affect the network efficiency of the device. If a device with an inefficient retransmission algorithm is found, NerveCenter sends a 7006 Inform to the platform.

Note The most efficient TCP retransmission algorithm currently available is Van Jacobson's Algorithm.

Two additional alarms, TcpRetransAlg_LogToDB and TcpRetransAlg_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on TCP transmission algorithms can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

tcp

Trigger Generators. Below is a list of the objects that generate triggers:

- TcpRetransAlg (poll)

Events Sent to OpenView. Table 7-2 summarizes the events this behavior model sends to OpenView:

Table 7-2. Events Sent to OpenView by TcpRetransAlg

Event Name	Inform Number	Description
NC_Tcp_RetAlg	7006	The current retransmission algorithm on this device is inefficient.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/tcp_status
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

TcpRetransMon

Path: `\model\tcp_status\tcp_retrans_mon.mod`

Monitors the percentage of TCP retransmissions for a particular device. Retransmission occurs when messages use the TCP protocol and a message is not acknowledged. NerveCenter sends a 7007 Inform to the platform if a high number of retransmissions is detected.

Two additional alarms, TcpRetransMon_LogToDB and TcpRetransMon_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on TCP retransmissions can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

tcp

Trigger Generators. Below is a list of the objects that generate triggers:

- TcpRetransStatus (poll)

Events Sent to OpenView. Table 7-3 summarizes the events this behavior model sends to OpenView:

Table 7-3. Events Sent to OpenView by TcpRetransMon

Event Name	Inform Number	Description
NC_Tcp_Retrans	7007	The current number of retransmissions is excessive for this device.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
\model\tcp_status
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Poll conditions for TCP retransmission thresholds

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

Troubleshooting Models 8

The troubleshooting models reside under `models/troubleshooting`, and consist of:

Section	Description
InformConnectStatus on page 44	Monitors the status of NerveCenter's connection with the OpenView Platform Adapter (OVPA) as well as the Inform Acknowledgement Queue.

InformConnectStatus

Path: `/model/troubleshooting/informconnectstatus.mod`

This behavior model monitors the status of NerveCenter's connection with the OpenView Platform Adapter (OVPA) as well as the Inform Acknowledgement Queue.

Property. The following property must be in the property group for the nodes you want to monitor:

NO_PROP

Trigger Generators. Below is a list of the objects that generate triggers:

- InformConnectStatus (alarm)

Events Sent to OpenView. InformConnectStatus sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)

Vendor Models 9

The vendor models reside under `models/vendors`, and consist of:

Section	Description
Cisco router models	**Behavior models that monitor various attributes of Cisco routers.**
CiscoBufferUtilization on page 47	Monitors the free element buffers on Cicso routers to determine if their buffer availability is at acceptable levels.
CiscoCPUUtilization on page 49	Monitors the CPU utilization on Cisco routers.
CiscoCRCCheck on page 51	Monitors the CRC errors on Cisco router interfaces.
CiscoFanCheck on page 53	Monitors the fan status in a Cisco router.
CiscoMemGetCheck on page 55	Monitors the available free memory for Cisco routers.
CiscoPacketsDrop on page 57	Monitors packets on Cisco router interfaces. Also monitors when both inbound and outbound packets are being dropped during the poll periods.
CiscoPwrSupplyCheck on page 59	Monitors the power supply status in a Cisco router.
CiscoTempCheck on page 61	Monitors the temperature status in a Cisco router.
Compaq device models	**Behavior models that monitor various attributes of Compaq devices.**
CpqDaLogDrvStat on page 63	Detects logical drive status changes such as overheating, a bad connection, expanding, or a failure.
CpqNic on page 65	Detects when a Network Interface Card (NIC) has failed on a Compaq server. CpqNic also detects when the server has switched to another redundant NIC.
CpqDaPhyDrvThresh on page 66	Monitors the threshold of an IDA physical drive.
CpqScsiPhyStat on page 67	Monitors the physical SCSI drive status.
CpqUnknown on page 69	Logs all traps from Compaq agents that are not used in the CpqTraps trap mask.
CpqUps on page 70	Detects the failure and degradation of a computer's uninterruptible power supply (UPS) system.

Section	Description
SNMP Research SNMP v3 models	**Behavior models that use SNMP Research's SNMP v3 entity, CIAgent to monitor system resources.**
SnmpresCritAppStatus on page 72	Provides status of applications monitored by SNMP Research's CIAgent.
SnmpresDiskUsage on page 74	Provides status of file systems monitored by SNMP Research's CIAgent.
SnmpresGenericLogMonitor on page 76	Monitors log file events reported by SNMP Research's CIAgent.
SnmpresLoginMonitor on page 78	Monitors failed logins reported by SNMP Research's CIAgent.
SnmpresMemUsage on page 80	Monitors status of system memory.
SnmpresProcessorLoad on page 82	Monitors the processor load.
Wellfleet (Nortel) router models	**Behavior models that monitor various attributes of Wellfleet (Nortel) routers.**
WfBufferUtilization on page 84	Monitors the free packet buffers on Wellfleet (Nortel) routers to determine if their buffer availability is at acceptable levels.
WfFanCheck on page 86	Monitors the fan status in a Wellfleet (Nortel) router.
WfMemCheck on page 88	Monitors the available free memory for Wellfleet (Nortel) routers.
WfPwrSupplyCheck on page 90	Monitors the power supplies status in a Wellfleet (Nortel) router.
WfTempCheck on page 92	Monitors the temperature status in a Wellfleet (Nortel) router.

CiscoBufferUtilization

Path: `/model/vendors/cisco/cisco_buffer_utilization.mod`

Monitors the free element buffers on Cicso routers to determine if their buffer availability is at acceptable levels. If buffer availability is less than five percent for three poll intervals, then NerveCenter sends a 6005 Inform to the platform.

Two additional alarms, CiscoBufferUtilization_LogToDB and CiscoBufferUtilization_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on buffer utilization can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

lsystem

Trigger Generators. Below is a list of the objects that generate triggers:

- CiscoBuffersCheck (poll)

Events Sent to OpenView. Table 9-1 summarizes the events this behavior model sends to OpenView:

Table 9-1. Events Sent to OpenView by CiscoBufferutilization

Event Name	Inform Number	Description
NC_Cisco_Buffer	6005	The buffer utilization for this device has been high over a persistent period of time.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/cisco
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Poll intervals

- Poll conditions for buffer utilization thresholds
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

CiscoCPUUtilization

Path: `/model/vendors/cisco/cisco_cpu_utilization.mod`

This behavior model monitors the CPU utilization on Cisco routers. NerveCenter sends a 6001 Inform to the platform only when the utilization reaches a very busy state and remains there for one hour.

Two additional alarms, CiscoCpuUtilization_LogToDB and CiscoCpuUtilization_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on CPU utilization can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

lsystem

Trigger Generators. Below is a list of the objects that generate triggers:

- CiscoCpuCheck (poll)
- CiscoCpuUtilization (alarm)
- CiscoCpuUtilization_LogToDB (alarm)
- CiscoCpuUtilization_LogToFile (alarm)

Events Sent to OpenView. Table 9-2 summarizes the events this behavior model sends to OpenView:

Table 9-2. Events Sent to OpenView by CiscoCPUUtilization

Event Name	Inform Number	Description
NC_Cisco_Cpu	6001	The CPU utilization for this device has been high over a persistent period of time.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/cisco
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Poll intervals
- Poll conditions for CPU utilization thresholds
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

CiscoCRCCheck

Path: `/model/vendors/cisco/cisco_crc_check.mod`

This model monitors the CRC errors on Cisco router interfaces. A poll checks to see if an increase in the number of CRC errors is occurring. If the threshold is exceeded, then a faster poll continues to check the interface for two more intervals. If the interface continues to exhibit CRC errors, NerveCenter sends a 6009 Inform to the platform.

Two additional alarms, CiscoCRCCheck_LogToDB and CiscoCRCCheck_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on CRC errors can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

lifEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CiscoCRCPoll (poll)
- CiscoCRCPollFast (poll)

Events Sent to OpenView. Table 9-3 summarizes the events this behavior model sends to OpenView:

Table 9-3. Events Sent to OpenView by CiscoCRCCheck

Event Name	Inform Number	Description
NC_Cisco_CRC	6009	The amount of CRC errors for this interface has been high over a persistent period of time.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/cisco
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set counters for persistence

- Poll intervals
- Poll conditions for CRC error thresholds
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

CiscoFanCheck

Path: `/model/vendors/cisco/cisco_fan_check.mod`

This behavior model monitors the fan status in a Cisco router. If a trap or a poll shows a problem, the alarm transitions to the FanWarning state which then polls the fan status again for verification. If the problem persists, NerveCenter sends a 6006 Inform to the platform. Once the problem has been corrected, the alarm returns to ground.

Two additional alarms, CiscoFanCheck_LogToDB and CiscoFanCheck_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on fan conditions can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

ciscoEnvMonFanStatusEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CiscoFanPoll (poll)
- CiscoFanPollFast (poll)
- CiscoFanFailureTrap (mask)

Events Sent to OpenView. Table 9-4 summarizes the events this behavior model sends to OpenView:

Table 9-4. Events Sent to OpenView by CiscoFanCheck

Event Name	Inform Number	Description
NC_Cisco_Fan	6006	A fan failure has been detected for this device.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/cisco
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

CiscoMemGetCheck

Path: `/model/vendors/cisco/cisco_mem_get_check.mod`

This multi-alarm model monitors the available free memory for Cisco routers.

The CiscoMemGet alarm obtains the memory installed in a Cisco router. The alarm will poll routers every six hours to ensure the memory configuration has not changed. It also listens for warm and cold starts which causes the router to be polled in case memory has been installed or removed.

The CiscoMemCheck alarm checks to see if the problem persists over three polling intervals. The free memory is tested for: low (6-10%), very low (< 6%), and zero-free-memory available conditions. On each state a poll is run, passing the resulting data through a Perl subroutine which uses the total memory installed in the router—along with the free memory just polled—to determine the percentage available.

Two additional alarms, CiscoMemCheck_LogToDB and CiscoMemCheck_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on memory utilization can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

chassis (CiscoMemGet) and lsystem (CiscoMemCheck)

Trigger Generators. Below is a list of the objects that generate triggers:

- CiscoMemCheck (poll)
- CiscoMemGet (poll)
- CiscoMemGetFast (poll)
- ColdStart (mask)
- WarmStart (mask)
- CiscoMemCheck (Perl subroutine)

Events Sent to OpenView. Table 9-5 summarizes the events this behavior model sends to OpenView:

Table 9-5. Events Sent to OpenView by CiscoMemGetCheck

Event Name	Inform Number	Description
NC_Cisco_LoMem	6010	The free memory available for this device has been low over a persistent period of time.
NC_Cisco_VeryLoMem	6011	The free memory available for this device has been very low over a persistent period of time.

Table 9-5. Events Sent to OpenView by CiscoMemGetCheck (continued)

Event Name	Inform Number	Description
NC_Cisco_ZeroMem	6012	The free memory available for this device has been zero over a persistent period of time.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/cisco
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set counters for persistence
- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

CiscoPacketsDrop

Path: `/model/vendors/cisco/cisco_packets_drop.mod`

Looks for a significant number of dropped inbound, outbound, or both types of packets on Cisco router interfaces. It checks to see if the problem persists over three polling intervals. Also monitors when both inbound and outbound packets are being dropped during the poll periods.

Two additional alarms, CiscoPacketsDrop_LogToDB and CiscoPacketsDrop_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on packet errors can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

lifEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CiscoPktsDropCheck (poll)

Events Sent to OpenView. Table 9-6 summarizes the events this behavior model sends to OpenView:

Table 9-6. Events Sent to OpenView by CiscoPacketsDrop

Event Name	Inform Number	Description
NC_Cisco_InPkts	6002	The amount of dropped inbound packets for this interface has been high over a persistent period of time.
NC_Cisco_OutPkts	6003	The amount of dropped outbound packets for this interface been high over a persistent period of time.
NC_Cisco_BothPkts	6004	The amount of dropped in and out bound packets for this interface has been high over a persistent period of time.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/cisco
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set counters for persistence
- Poll intervals
- Poll conditions for input and output queue lengths
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

CiscoPwrSupplyCheck

Path: `/model/vendors/cisco/cisco_pwr_supply_check.mod`

This behavior model monitors the power supply status in a Cisco router. If a trap or a poll shows a problem, the alarm transitions to the PowerSupplyWarning state which then polls the power supply status again for verification. If there is a problem, NerveCenter sends a 6007 Inform to the platform. Once the problem has been corrected, the alarm returns to ground.

Two additional alarms, Cisco`PwrSupplyCheck`_LogToDB and Cisco`PwrSupplyCheck`_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on power supply status can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

ciscoEnvMonSupplyStatusEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CiscoPwrSupplyPoll (poll)
- CiscoPwrSupplyPollFast (poll)
- CiscoPwrSupplyFailureTrap (mask)

Events Sent to OpenView. Table 9-7 summarizes the events this behavior model sends to OpenView:

Table 9-7. Events Sent to OpenView by CiscoPwrSupplyCheck

Event Name	Inform Number	Description
NC_Cisco_PowerSupply	6007	A power supply failure has been detected for this device.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/cisco
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

CiscoTempCheck

Path: `/model/vendors/cisco/cisco_temp_check.mod`

This behavior model monitors the temperature status in a Cisco router. If a trap or a poll shows a problem, the alarm transitions to the TempWarning state which then polls the temperature status again for verification. If there is a problem, NerveCenter sends a 6008 Inform to the platform. Once the problem has been corrected, the alarm returns to ground.

Two additional alarms, CiscoTempCheck_LogToDB and CiscoTempCheck_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on temperature status can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

ciscoEnvMonTemperatureStatusEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CiscoTempPoll (poll)
- CiscoTempPollFast (poll)
- CiscoTempTrap (mask)

Events Sent to OpenView. Table 9-8 summarizes the events this behavior model sends to OpenView:

Table 9-8. Events Sent to OpenView by CiscoTempCheck

Event Name	Inform Number	Description
NC_Cisco_Temp	6008	An over-temperature condition has been detected for this device.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

`/model/vendors/cisco`

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

CpqDaLogDrvStat

Path: `/model/vendors/compaq/compaq_logical_drive.mod`

This behavior model detects logical drive status changes such as overheating, a bad connection, expanding, or a failure.

Property. The following property must be in the property group for the nodes you want to monitor:

cpqDaLogDrvEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CpqDaLogDrvBadConn (poll)
- CpqDaLogFailed (poll)
- CpqDaLogOK (poll)
- CpqTraps (mask)

Events Sent to OpenView. Table 9-9 summarizes the events this behavior model sends to OpenView:

Table 9-9. Events Sent to OpenView by CpqDalogDrvStat

Event Name	Inform Number	Description
NC_CpqLogOverheat	1208	This inform event is triggered by the CpqDaLogDrvStat alarm to indicate that a an overheat indication has been detected on the physical drive.
NC_CpqLogShutDown	1209	This inform event is triggered by the CpqDaLogDrvStat alarm to indicate that a physical drive has shutdown.
NC_CpqLogFailed	1210	This inform event is triggered by the CpqDaLogDrvStat alarm to indicate that the log device has failed.
NC_CpqLogBadConn	1211	This inform event is triggered by the CpqDaLogDrvStat alarm to indicate that the connection to the data log device has been lost.
NC_CpqLogOK	1212	This inform event is triggered by the CpqDaLogDrvStat alarm to indicate that the failed physical drive has recovered.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/compaq
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- SMTP mail recipient
- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter

CpqNic

Path: `/model/vendors/compaq/compaq_nic.mod`

This behavior model detects when a Network Interface Card (NIC) has failed on a Compaq server. It also detects when the server has switched to another redundant NIC. When a failure is detected, NerveCenter extracts from the varbind information which slot contains the failed card. NerveCenter then sends this information in an email as well as informs the network management platform.

Property. The following property must be in the property group for the nodes you want to monitor:

compaq

Trigger Generators. Below is a list of the objects that generate triggers:

- CpqTraps (mask)

Events Sent to OpenView. Table 9-10 summarizes the events this behavior model sends to OpenView:

Table 9-10. Events Sent to OpenView by CpqNic

Event Name	Inform Number	Description
NC_CpqNicFailed	1203	This inform event is triggered by the CpqNic alarm to indicate that a network interface card has failed.
NC_CpqNicOK	1204	This inform event is triggered by the CpqNic alarm to indicate that network interface card has recovered.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/compaq
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- SMTP mail recipient
- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter

CpqDaPhyDrvThresh

Path: `/model/vendors/compaq/compaq_physical_drive.mod`

This behavior model monitors the threshold of an IDA physical drive. When NerveCenter receives a threshold trap, it polls the various attributes that track whether the device has exceeded its threshold. If the device has in fact exceeded the threshold, then the alarm transitions to PreFailure. If not, a diagnostic test can be run on the unit.

Property. The following property must be in the property group for the nodes you want to monitor:

cpqDaPhyDrvThrEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CpqDaPhyDrvThresh (poll)
- CpqDaPhyDrvThresh2 (poll)
- CpqTraps (mask)

Events Sent to OpenView. Table 9-11 summarizes the events this behavior model sends to OpenView:

Table 9-11. Events Sent to OpenView by CpqDaPhyDrvThresh

Event Name	Inform Number	Description
NC_CpqDaPhyDrvFail	1214	This inform event is triggered by the CpqDaPhyDrvThresh alarm to indicate that the threshold of the device has been exceeded.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

`/model/vendors/compaq`

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- SMTP mail recipient
- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter

CpqScsiPhyStat

Path: `/model/vendors/compaq/compaq_scsi_drive.mod`

This behavior model monitors the physical SCSI drive status. It checks to see if the device has failed, has a bad cable connection, or has been removed.

Property. The following property must be in the property group for the nodes you want to monitor:

cpqScsiPhyDrvEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CpqScsiPhyBadConn (poll)
- CpqScsiPhyFailed (poll)
- CpqScsiPhyOK (poll)
- CpqTraps (mask)

Events Sent to OpenView. Table 9-12 summarizes the events this behavior model sends to OpenView:

Table 9-12. Events Sent to OpenView by CpqScsiPhyStat

Event Name	Inform Number	Description
NC_CpqPhyOK	1205	This inform event is triggered by the CpqNic alarm to indicate that the physical SCSI drive has recovered.
NC_CpqPhyFail	1206	The inform event is triggered by the CpqNic alarm to indicate that the physical SCSI drive has recovered.
NC_CpqPhyBadConn	1207	This inform event is triggered by the CpqPhyBadConn alarm to indicate that a bad connection has been detected on the physical SCSI drive.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/compaq
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- SMTP mail recipient
- Log file pathname
- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter

CpqUnknown

Path: `/model/vendors/compaq/compaq_unknown_traps.mod`

This behavior model logs all traps from Compaq agents that are not used in the CpqTraps trap mask. This will allow the user to see which Compaq traps could be used in future behavior models.

Property. The following property must be in the property group for the nodes you want to monitor:

compaq

Trigger Generators. Below is a list of the objects that generate triggers:

- CpqTraps (mask)

Events Sent to OpenView. CpqUnknown sends no events to OpenView.

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)

CpqUps

Path: `/model/vendors/compaq/compaq_ups.mod`

This behavior model detects the failure and degradation of a computer’s uninterruptible power supply (UPS) system. If a failure is detected via a trap, NerveCenter first takes a quick poll to make sure it is not a power surge. If it is not a surge, the alarm transitions in to the ACLineFail state. An e-mail is sent notifying the administrator that the UPS has lost AC power and how much time is left in the UPS. It will also include whether or not the system attached to the UPS will perform an automatic shutdown when the UPS is low on power.

Property. The following property must be in the property group for the nodes you want to monitor:

cpqUpsBasic

Trigger Generators. Below is a list of the objects that generate triggers:

- CpqUpsLowPower (poll)
- CpqUpsOK (poll)
- CpqUpsOKQuick (poll)
- CpqTraps (mask)
- CpqUps (alarm)

Events Sent to OpenView. Table 9-13 summarizes the events this behavior model sends to OpenView:

Table 9-13. Events Sent to OpenView by CpqUps

Event Name	Inform Number	Description
NC_CpqUpsFail2	1200	This inform event is triggered by the CpqUps alarm to indicate that the AC power to the interruptible power supply has failed.
NC_CpqUpsOK	1201	This inform event is triggered by the CpqUps alarm to indicate that the AC power to the interruptible power supply has been restored.
NC_CpqUpsShutDown	1202	This inform event is triggered by the CpqUps alarm to indicate that the interruptible power supply has been shutdown.
NC_CpqUpsLow	1213	This inform event is triggered by the CpqUps alarm to indicate that a low battery level on the interruptible power supply has been shut down.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/compaq
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter

SnmpresCritAppStatus

Path: `/model/vendors/snmp_research/snmpres_crit_app_status.mod`

This behavior model provides the status of the applications monitored by SNMP Research's CIAgent. When an application goes down and remains down for a specified time (the default is ten minutes), NerveCenter sends a 110031 Inform to the platform. When an application comes back up, the alarm returns to ground and sends a 120031 Inform to the platform. The nodes hosting the monitored applications must be associated with a property group that contains the property critAppProcEntry.

Property. The following property must be in the property group for the nodes hosting the applications you want to monitor:

critAppProcEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CritAppDown (mask or poll)
- CritAppUp (mask or poll)
- CritAppDownFast (poll)
- CritAppUpFast (poll)
- CritAppAdminDownFast (poll)
- CritAppDownNotify (alarm)

Events Sent to OpenView. Table 9-14 summarizes the events this behavior model sends to OpenView:

Table 9-14. Events Sent to OpenView by SnmpresCritAppStatus

Event Name	Inform Number	Description
NC_Crit_App_Down	110031	CritAppDown trap received from CIAgent, or polling determined that the application is down.
NC_Crit_App_Up	120031	CritAppUp trap received from CIAgent, or polling determined that the application is up.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/snmp_research
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Change Inform to Inform Platform for Netcool, TME, Unicenter

CIAgent Customization. The following is a list of sample CIAgent configurations that you might want to customize to suit the requirements of your site:

- On Windows, the configuration file for the critical applications agent is: `C:\etc\srconf\agt\critagt.cnf`

  Below is a sample entry for monitoring the DNS service:

  ```
  critAppProcEntry  2 DNS "net start dns" "net stop dns" 1 true  3000
  true false \ true
  ```

- On UNIX, the configuration file for the critical applications agent is:
- `/etc/srconf/agt/critagt.cnf`

  Below is a sample entry for monitoring sendmail:

  ```
  critAppProcEntry  1 sendmail "/etc/rc2.d/S88sendmail start"
  "/etc/rc2.d/S88sendmail stop" 1 \
  true  3000 true true true
  ```

The *true* parameter in these examples tells CIAgent to restart the application if it fails. Make this parameter false when you do not want CIAgent to restart the application.

SnmpresDiskUsage

Path: `/model/vendors/snmp_research/snmpres_disk_usage.mod`

This behavior model informs the status of the file systems monitored by SNMP Research's CIAgent. When the disk usage of a file system exceeds a threshold (90 percent by default) and remains above the threshold for a specified time (the default is fifteen minutes), NerveCenter sends a 110041 Inform to the platform. When the disk usage falls below the threshold, the alarm returns to ground and sends a 120041 Inform to the platform. The nodes that have their file systems monitored must be associated with a property group that contains the property siFsrEntry.

Property. The following property must be in the property group for the nodes hosting the applications you want to monitor:

siFsEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- DiskUsageOk (mask or poll)
- HighDiskUsage (mask or poll)
- DiskUsageOkFast (poll)
- HighDiskUsageFast (poll)
- DiskUsageNotify (alarm)

Events Sent to OpenView. Table 9-15 summarizes the events this behavior model sends to OpenView:

Table 9-15. Events Sent to OpenView by SnmpresDiskUsage

Event Name	Inform Number	Description
NC_Disk_Usage_High	110041	Disk usage over 90 percent.
NC_Disk_Usage_Ok	120041	Disk usage dropped under 90 percent.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/snmp_research
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Change Inform to Inform Platform for Netcool, TME, Unicenter

SnmpresGenericLogMonitor

Path: `/model/vendors/snmp_research/snmpres_generic_log_monitor.mod`

This behavior model monitors log file events reported by SNMP Research's CIAgent. When the amount of events reaches the threshold (three by default) within a specified time (the default is fifteen minutes), NerveCenter sends a 110081 Inform to the platform. If the time expires before the threshold is reached, the alarm is grounded and inform 120081 is sent. The nodes monitored must be associated with a property group that contains the property siLogEntry.

Property. The following property must be in the property group for the nodes hosting the applications you want to monitor:

siLogEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- LogFileEvent (mask)
- LogFileEventWarning (alarm)
- LogFileEventTimer (alarm)

Events Sent to OpenView. Table 9-16 summarizes the events this behavior model sends to OpenView:

Table 9-16. Events Sent to OpenView by SnmpresGenericLogMonitor

Event Name	Inform Number	Description
NC_Log_File_Event	110081	A logged event occurred three times in 15 minutes.
NC_Log_File_Event	120081	A logged event occurred less than three times in 15 minutes.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/snmp_research
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers and counters for persistence

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Configure the mask to instantiate different alarms according to the log entry received

CIAgent Customization. The following is a list of sample CIAgent configurations that you might want to customize to suit the requirements of your site:

- On Windows, there are three log files: Application, Security, and System. In order to monitor entries within them, you must not use the path to the file in the configuration entry. Instead, use “Event Log: *log filename*: *source*”, where *log filename* is the name of the log file; that is, Security, and the *source* is the application or service generating the log entry. Below are two sample entries; one matches a “failed to renew” string in a Dhcp event in the System log; the other matches “application error” strings in DrWatson events in the Application log:

  ```
  siLogEntry  1 System "Event Log: System: Dhcp" "failed to renew" 2 4 0
  1 - 1 \ 2031616 0 - 2 10 2 System 1
  ```

  ```
  siLogEntry  2 Application "Event Log: Application: DrWatson"
  "application error" 2 4 0 \ 0 - 1 524288 0 - 2 15 2 Application 1
  ```

  The default path to the CIAgent log file monitoring configuration file on Windows is:

  ```
  C:\etc\srconf\agt\logagt.cnf
  ```

- On UNIX, the location of the log files varies for every platform and application. Below are two sample entries for monitoring the syslog files on Solaris and on HP-UX, respectively:

  ```
  siLogEntry  1 Syslog /var/adm/messages "login: *pam_authenticate:
  *error" 2 10 \ 200 25 - 1 8233138 0 - 2 15 2 System 1
  ```

  ```
  siLogEntry  3 Syslog /var/adm/syslog/syslog.log "sendmail: *SYSERR*
  exiting" 2 10 0 \ 0 - 1 8233138 0 - 2 15 2 system 1
  ```

  The default path to the CIAgent log file monitoring configuration file on UNIX is:

  ```
  /etc/srconf/agt/logagt.cnf
  ```

Tip Verify that the appropriate message level for the events you are trying to monitor is configured in the `/etc/syslog.conf` file.

SnmpresLoginMonitor

Path: `/model/vendors/snmp_research/snmpres_login_monitor.mod`

This behavior model monitors failed logins reported by SNMP Research's CIAgent. When the number of attempts reaches the threshold (three by default) within a specified time (the default is five minutes), NerveCenter sends a 110071 Inform to the platform. If the time expires before the attempt threshold is reached, the alarm is grounded. The nodes monitored must be associated with a property group that contains the property siLogEntry.

Property. The following property must be in the property group for the nodes hosting the applications you want to monitor:

siLogEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- FailedLogin (mask)
- LoginWarning (alarm)
- LoginWarningTimer (alarm)

Events Sent to OpenView. Table 9-17 summarizes the events this behavior model sends to OpenView:

Table 9-17. Events Sent to OpenView by SnmpresLoginMonitor

Event Name	Inform Number	Description
NC_Login_Failed	110071	Login attempted three times in five minutes.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/snmp_research
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers and counters for persistence
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Return alarm to ground according to your security policy, after a timer expires or manually

CIAgent Customization. The following is a sample CIAgent configuration that you might want to customize to suit the requirements of your site:

On UNIX, the location of the log files varies for every platform and application. Below is a sample entry for monitoring the `syslog` file on Solaris:

```
siLogEntry  1 Syslog /var/adm/messages "login: *pam_authenticate:
*error" 2 10 \ 200 25 - 1 8233138 0 - 2 15 2 System 1
```

The default path to the CIAgent log file monitoring configuration file on UNIX is:

```
/etc/srconf/agt/logagt.cnf
```

Tip Verify that the appropriate message level for the events you are trying to monitor is configured in the `/etc/syslog.conf` file.

SnmpresMemUsage

Path: `/model/vendors/snmp_research/snmpres_mem_usage.mod`

This behavior model monitors the status of the system memory. When the memory utilization exceeds the threshold (90 percent by default) and remains above the threshold for a specified time (the default is fifteen minutes), NerveCenter sends a 110051 Inform to the platform. When the memory utilization falls below the threshold, the alarm returns to ground and sends a 120051 Inform to the platform. The nodes that have their memory utilization monitored must be associated with a property group that contains the property hrStrorageEntry.

Property. The following property must be in the property group for the nodes hosting the applications you want to monitor:

hrStrorageEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- MemUsageOk (poll)
- HighMemUsage (poll)
- MemUsageOkFast (poll)
- HighMemUsageFast (poll)
- MemUsageNotify (alarm)

Events Sent to OpenView. Table 9-18 summarizes the events this behavior model sends to OpenView:

Table 9-18. Events Sent to OpenView by SnmpresMemUsage

Event Name	Inform Number	Description
NC_Mem_Usage_High	110051	Memory utilization exceeded 90 percent.
NC_Mem_Usage_Ok	120051	Memory utilization dropped under 90 percent.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/snmp_research
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Change Inform to Inform Platform for Netcool, TME, Unicenter

SnmpresProcessorLoad

Path: `/model/vendors/snmp_research/snmpres_processor_load.mod`

This behavior model monitors the processor load. When the processor load reaches the threshold (90 percent by default) and remains above the threshold for a specified time (the default is fifteen minutes), NerveCenter sends a 110061 Inform to the platform. When the processor load falls below the threshold, the alarm returns to ground and sends a 120061 Inform to the platform. The nodes that have their processors monitored must be associated with a property group that contains the property hrProcessorEntry.

Property. The following property must be in the property group for the nodes hosting the applications you want to monitor:

hrProcessorEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- CPUok (poll)
- CPUokFast (poll)
- CPUbusy (poll)
- CPUbusyFast (poll)
- CPUbusyNotify (alarm)

Events Sent to OpenView. Table 9-19 summarizes the events this behavior model sends to OpenView:

Table 9-19. Events Sent to OpenView by SnmpresProcessorLoad

Event Name	Inform Number	Description
NC_CPU_Busy	110061	Processor load reached 90 percent.
NC_CPU_Ok	120061	Processor load dropped under 90 percent.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/snmp_research
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set timers for persistence
- Change Inform to Inform Platform for Netcool, TME, Unicenter

WfBufferUtilization

Path: `/model/vendors/wellfleet/wf_buffer_utilization.mod`

Monitors the free packet buffers on Wellfleet (Nortel) routers to determine if their buffer availability is at acceptable levels. If buffer availability is less than five percent for three poll intervals, then NerveCenter sends a 6505 Inform to the platform.

Two additional alarms, WfBufferUtilization_LogToDB and WfBufferUtilization_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on buffer availability can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

wfKernelEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- WfBuffersCheck (poll)

Events Sent to OpenView. Table 9-20 summarizes the events this behavior model sends to OpenView:

Table 9-20. Events Sent to OpenView by WfBufferUtilization

Event Name	Inform Number	Description
NC_Wellfleet_Buffer	6505	The buffer utilization for this device has been high over a persistent period of time.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

`/model/vendors/wellfleet`

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set counters for persistence
- Poll intervals

- Poll conditions for buffer utilization thresholds
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

WfFanCheck

Path: `/model/vendors/wellfleet/wf_fan_check.mod`

Monitors the fan status in a Wellfleet (Nortel) router. If a poll shows a problem, the alarm transitions to the FanWarning state which then polls the fan status again for verification. If there is a problem, NerveCenter sends a 6506 Inform to the platform. Once the problem has been corrected, the alarm returns to ground.

Two additional alarms, WfFanCheck_LogToDB and WfFanCheck_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on fan status can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

wfKernelEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- WfFanPoll (poll)
- WfFanPollFast (poll)

Events Sent to OpenView. Table 9-21 summarizes the events this behavior model sends to OpenView:

Table 9-21. Events Sent to OpenView by WfFanCheck

Event Name	Inform Number	Description
NC_Wellfleet_Fan	6506	A fan failure has been detected for this device.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

`/model/vendors/wellfleet`

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

WfMemCheck

Path: `/model/vendors/wellfleet/wf_mem_check.mod`

This model monitors the available free memory for Wellfleet (Nortel) routers. NerveCenter checks to see if the conditions persist over three polling intervals. The free memory is tested for the following conditions: low (6-10%), very low (< 6%), and zero-free-memory available.

Two additional alarms, WfMemCheck_LogToDB and WfMemCheck_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on memory availability can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

wfKernelEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- WfMemCheck (poll)

Events Sent to OpenView. Table 9-22 summarizes the events this behavior model sends to OpenView:f

Table 9-22. Events Sent to OpenView by WfMemCheck

Event Name	Inform Number	Description
NC_Wellfleet_LoMem	6510	The free memory available for this device has been low over a persistent period of time.
NC_Wellfleet_VeryLoMem	6511	The free memory available for this device has been very low over a persistent period of time.
NC_Wellfleet_ZeroMem	6512	The free memory available for this device has been zero over a persistent period of time.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/wellfleet
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Set counters for persistence
- Poll intervals
- Poll conditions for memory utilization thresholds
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

WfPwrSupplyCheck

Path: `/model/vendors/wellfleet/wf_pwr_supply_check.mod`

This behavior model monitors the power supplies status in a Wellfleet (Nortel) router. If a poll shows a problem with any of the power supplies, the alarm transitions to the PowerSupplyWarning state which then polls the power supply status again for verification. If there is a problem, NerveCenter sends a 6507 Inform to the platform. Once the problem has been corrected, the alarm returns to ground.

Two additional alarms, WfPwrSupplyCheck_LogToDB and WfPwrSupplyCheck_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on power supply status can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

wfHwBase

Trigger Generators. Below is a list of the objects that generate triggers:

- WfPwrSupplyPoll (poll)
- WfPwrSupplyPollFast (poll)

Events Sent to OpenView. Table 9-23 summarizes the events this behavior model sends to OpenView:

Table 9-23. Events Sent to OpenView by WfPwrSupplyCheck

Event Name	Inform Number	Description
NC_Wellfleet_PowerSupply	6507	A power supply failure has been detected for this device.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

`/model/vendors/wellfleet`

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It’s a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals
- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

WfTempCheck

Path: `/model/vendors/wellfleet/wf_temp_check.mod`

This behavior model monitors the temperature status in a Wellfleet (Nortel) router. If a poll shows a problem, the alarm transitions to the TempWarning state which then polls the temperature status again for verification. If there is a problem, NerveCenter sends a 6508 Inform to the platform. Once the problem has been corrected, the alarm returns to ground.

Two additional alarms, WfTempCheck_LogToDB and WfTempCheck_LogToFile, are part of this model. When you enable these alarms, they perform a Log To Database or a Log to File alarm action, respectively, on each transition so that reports on temperature conditions can be produced.

Note The LogToDB alarm, works on Windows only.

Property. The following property must be in the property group for the nodes you want to monitor:

wfKernelEntry

Trigger Generators. Below is a list of the objects that generate triggers:

- WfTempPoll (poll)
- WfTempPollFast (poll)

Events Sent to OpenView. Table 9-24 summarizes the events this behavior model sends to OpenView:

Table 9-24. Events Sent to OpenView by WfTempCheck

Event Name	Inform Number	Description
NC_Wellfleet_Temp	6508	An over-temperature condition has been detected for this device.

For complete information about events sent to OpenView, see the `trapd.conf.txt` file shipped with these behavior models residing in:

```
/model/vendors/wellfleet
```

For more information about integrating NerveCenter with OpenView, or the other NerveCenter-supported network manager platforms, refer to the book *Integrating NerveCenter with a Network Management Platform*.

Optional Customization. Below is a list of behavior model settings that you might want to customize to suit the requirements of your site.

Tip It's a good practice to rename the behavior model objects that you modify. Whenever you import a model, NerveCenter will overwrite objects with the same name.

- Poll intervals

- Change Inform to Inform Platform for Netcool, TME, Unicenter
- Log To File alarm action options
  (file pathname, log action variables, verbose or non-verbose output)
- Log To Database alarm action options
  (log identity number and log action variables)

Index

A

application status behavior model 72
Authentication behavior model 4

B

behavior models
- authentication failures 4
- Cisco router buffer 47
- Cisco router CPU 49
- Cisco router CRC 51
- Cisco router fan 53
- Cisco router memory 55
- Cisco router packets 57
- Cisco router power supply 59
- Cisco router temperature 61
- Compaq logical drive 63
- Compaq network interface card (NIC) 65
- Compaq physical drive 66
- Compaq SCSI drive 67
- Compaq uninterruptible power supply (UPS) 70
- Compaq unknown traps 69
- discovery of nodes (NT) 10
- gateways 7
- Inform Acknowledgement Queue connection 44
- interface attribute values 8
- IP sweep (NT) 10
- log ifEntry to database 14
- node routing 12
- OVPA connection 44
- ping sweep (NT) 10
- routing monitoring 12
- SNMP Research application status 72
- SNMP Research failed login events 78
- SNMP Research file system status 74
- SNMP Research log file events 76
- SNMP Research memory status 80
- SNMP Research processor load 82
- SNMP v3 72, 74, 76, 78, 80, 82
- Wellfleet (Nortel) router buffer 84
- Wellfleet (Nortel) router fan 86
- Wellfleet (Nortel) router memory 88
- Wellfleet (Nortel) router power supply 90
- Wellfleet (Nortel) router temperature 92

buffer (router) behavior models 47, 84

C

CIAgent behavior models 72, 74, 76, 78, 80, 82
Cisco router behavior models 47, 49, 51, 53, 55, 57, 59, 61
CiscoBufferUtilization behavior model 47
CiscoCPUUtilization behavior model 49
CiscoCRCCheck behavior model 51
CiscoFanCheck behavior model 53
CiscoMemGetCheck behavior model 55
CiscoPacketsDrop behavior model 57
CiscoPwrSupplyCheck behavior model 59
CiscoTempCheck behavior model 61
Compaq behavior models 63, 65, 66, 67, 69, 70
CpqDaLogDrvStat behavior model 63
CpqDaPhyDrvThresh behavior model 66
CpqNic behavior model 65
CpqScsiPhyStat behavior model 67
CpqUnknown behavior model 69
CpqUps behavior model 70
CPU (router) behavior models 49
CRC (router) behavior models 51

D

dbi behavior model 14
discovery behavior model 10
drives
- IDA drive (Compaq) behavior model 66
- logical drive (Compaq) behavior model 63
- physical drive (Compaq) behavior model 66
- SCSI drive (Compaq) behavior model 67

F

failed login events behavior model 78
fan (router) behavior models 53, 86
file system status behavior model 74

G

gateway behavior model 7
GatewayList behavior model 7

I

IDA drive (Compaq) behavior model 66
IfData_LogToFile behavior model 8
ifEntry_LogToDBI behavior model 14
Inform Acknowledgement Queue behavior model 44

InformConnectStatus behavior model 44
interface attribute values behavior model 8
IPSweep (NT) behavior model 10
IPSWEEP.EXE 10

L
load, processor behavior model 82
log file events behavior model 76
logical drive behavior model 63

M
memory (router) behavior models 55, 88
memory status behavior model 80

N
network authentication failures behavior model 4
network interface card (NIC) behavior model 65
node routing behavior model 12
Nortel *See* Wellfleet router behavior models

O
OVPA connection behavior model 44

P
packets (router) behavior models 57
physical drive behavior model 66
ping sweep behavior model 10
power supply (Compaq) behavior models 70
power supply (router) behavior models 59
power supply behavior models 90
processor load behavior model 82

R
routers
- Cisco buffer behavior model 47
- Cisco CPU behavior model 49
- Cisco fan behavior model 53
- Cisco memory behavior model 55
- Cisco packets behavior model 57
- Cisco power supply behavior model 59
- Cisco temperature behavior model 61
- CiscoCRC behavior model 51
- Wellfleet (Nortel) buffer behavior model 84
- Wellfleet (Nortel) fan behavior model 86
- Wellfleet (Nortel) memory behavior model 88
- Wellfleet (Nortel) power supply behavior model 90
- Wellfleet (Nortel) temperature behavior model 92

RoutingMon behavior model 12

S
SCSI drive behavior model 67
SNMP Research behavior models 72, 74, 76, 78, 80, 82
SNMP v3 behavior models 72, 74, 76, 78, 80, 82
SnmpresCritAppStatus behavior model 72
SnmpresDiskUsage behavior model 74
SnmpresGenericLogMonitor behavior model 76
SnmpresLoginMonitor behavior model 78
SnmpresMemUsage behavior model 80
SnmpresProcessorLoad behavior model 82
status, application behavior model 72
status, file system behavior model 74
status, memory behavior model 80

T
temperature (router) behavior models 61, 92
traps
- behavior model for unknown traps (Compaq) 69

U
uninterruptible power supply (UPS) behavior model 70

W
Wellfleet (Nortel) router behavior models 84, 86, 88, 90, 92
WfBufferUtilization behavior model 84
WfFanCheck behavior model 86
WfMemCheck behavior model 88
WfPwrSupplyCheck behavior model 90
WfTempCheck behavior model 92