CA Spectrum®

Device Management Reference Guide

Release 9.4

ca technologies

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: Getting Started

This guide introduces CA Spectrum Device Management documentation for the following devices (presented alphabetically):

- Chapter 2: AM Communications
- Chapter 3: Ceterus Universal
- Chapter 4: Cheetah Gateway
- Chapter 5: HP BladeSystem c-Class Certification
- Chapter 6: Juniper M Series
- Chapter 7: Netscreen Firewall
- Chapter 8: Nortel Contivity VPN Switches

Chapter 2: AM Communications

This section introduces the CA Spectrum Device Management documentation for the AM Communications Integration.

This section contains the following topics:

Supported Devices

AM Communications develops network management products for non-SNMP broadband components. They monitor RF (Radio Frequency), HFC (Hybrid FIber Coax) components. The NetMentor software package of Cheetah, with optional SNMP Agent Module, converts their proprietary events into SNMPv1/v2 alarms and traps. This management module uses the Generic Southbound Application Gateway integration to provide a place for trap reception and event creation.

This management module supports the Omni2000 Proxy Agent. Omni2000 Proxy Agent is the HFC component monitoring solution of AM Communication.

The CA Spectrum Model

No specific AM Communications model types are created. The Southbound Gateway provides the model types EventAdmin and EventModel. These model types are used to manage the information that the Omni2000 Proxy Agent sends to CA Spectrum.

EventAdmin is a container model type that is used to represent the Omni2000 Proxy Agent. EventModels represent unique sources of trap information that the Omni2000 Proxy Agent passes to CA Spectrum. EventModels are automatically placed in a topology view that you can access by drilling down from the EventAdmin model. These icons do not show any connectivity with one another because they represent an event source, not necessarily a physical device or component.

When the EventAdmin model receives a trap from the Omni2000 Proxy Agent, it maps the trap to a CA Spectrum event. It then sends the event to the appropriate EventModel for processing. If an EventModel that represents the unique source of trap information does not exist, it is automatically created.

The *SouthBound Gateway Toolkit Guide* contains instructions for creating an EventAdmin model. Use the EventAdmin model to represent the AM Communications management application.

When you create this model, select a Manager Name of Omni2000.

Traps, Events, and Alarms

This section describes how the AM Communication Integration sends the traps. It also describes how the EventAdmin and EventModels process and manage these traps.

As the Omni2000 Proxy Agent sends traps to CA Spectrum, the EventAdmin model receives them, and maps them to a CA Spectrum event. These events are sent to an EventModel that represents the trap source. The value of the NEModelNumber variable binding that is sent in the trap identifies the trap source. This variable binding is from the AMC-MIB**.** If an EventModel representing the trap source does not exist, it automatically is created.

When the EventModel receives the event, it is processed and can be used to create or clear an alarm. The following table displays how each trap is mapped to a CA Spectrum event and how the event is processed.

trap	alarm	event code	description
NewNEFound		0x3eb0001	HFC Proxy detected new Network Element.
Communicatio-nsStatus		0x3eb0002	HFC Proxy lost or restored communication with Network Element.
Configuration Change	orange	ox3eb0003	Configuration of a single variable of any type was changed (via any interface).
StatusChange		0x3eb0004	An active alarm was cleared.
Alarm	orange	0x3eb0005	An ALARM is detected by a proxy agent.
ToBeSendQue-ueOverflow	orange	0x3eb0006	SNMP agent's TrapToBeSendQueue is full.

trap	alarm	event code	description
NewNELost	orange	0x3eb0007	HFC Proxy Detected New Network Element Lost.

Chapter 3: Ceterus Universal

This section describes common deployment scenarios for the Ceterus Universal Transport System devices and how to model them in CA Spectrum.

This section contains the following topics:

Trap Processing

You can configure the Remote Ceterus device to forward traps through its EOC channel to the Local device. In this configuration, the Local device acts as a gateway and forwards these traps. For more information about this feature, see the Ceterus documentation.

You can also configure the Remote device with an SNMP target IP address. In this configuration, the device sends traps through its management port.

If both of these capabilities are configured simultaneously, CA Spectrum receives duplicate traps. The Ceterus management module is designed to handle this case. It evaluates the incoming Ceterus traps and asserts those traps on the most appropriate CA Spectrum device model. The management module selects the appropriate model by comparing the Ceterus device community string in the trap with the value of the device sysName.

Important: CA Spectrum relies on the community name of the device to make this determination. As a result, the community name and the sysName must be synchronized. Polling of sysName occurs every 5 minutes by default. Changing the TID can affect the handling of the trap until sysName has been properly updated through a poll. When an administrator changes the TID (sysName) on a given Ceterus device from “Device A” to “Device B,” the device sends traps to that model. In this case, CA Spectrum can no longer process the traps. Trap processing does not recommence until the sysName is updated (up to a maximum of 5 minutes by default).

Chapter 4: Cheetah Gateway

This section describes CA Spectrum support for monitoring Cheetah™ network management products.

This section contains the following topics:

Supported Devices

Cheetah™ products, including CheetahNet™ (formerly NetMentor™) are network management products for non-SNMP broadband components. They monitor RF (Radio Frequency), and HFC (Hybrid Fiber Coax) components. The CheetahNet/NetMentor software package, with an optional SNMP Agent Module, converts their proprietary events into SNMPv1/v2 alarms and traps. This management module uses the CA Spectrum Southbound Gateway integration to enable traps to be received and events to be created in CA Spectrum.

This management module provides an integration between the CheetahNet/NetMentor management application, including the SNMP agent module, and CA Spectrum. This integration can report events on the following types of HFC devices:

- Power Supply
- Amplifier
- Line Monitor
- Test Point
- Fiber Node
- HEFiber

The CA Spectrum Model

No specific Cheetah model types are created. The Southbound Gateway provides the EventAdmin and EventModel model types. These model types are used to manage the information that NetMentor sends to CA Spectrum.

The EventAdmin is a container model type that is used to represent the NetMentor management application. EventModels represent unique sources of trap information that the CheetahNet/NetMentor application passes to CA Spectrum. EventModels are automatically placed in a topology view that can be accessed by drilling down from the EventAdmin model. These icons do not show any connectivity with one another because they represent an event source, not necessarily a physical device or component.

When the EventAdmin model receives a trap from the CheetahNet/NetMentor application, it maps the trap to a CA Spectrum event. The EventAdmin also sends the event to the appropriate EventModel for processing. If an EventModel that represents the unique source of trap information does not exist, it is automatically created.

Creating the EventAdmin Model

The *SouthBound Gateway Toolkit Guide* contains instructions for creating an EventAdmin Model. Use the EventAdmin model to represent the CheetahNet/NetMentor management application. When you create this model, select a Manager Name of NetMentor.

Traps, Events, and Alarms

This section describes how the EventAdmin and EventModel process and manage traps that are sent by the CheetahNet/NetMentor integration.

As CheetahNet/NetMentor sends traps to CA Spectrum, the EventAdmin model receives these traps and maps them to a CA Spectrum event. These events are sent to an EventModel that represents the trap source. The value of the **CNAlarmResource** and the **CNAlarmSubResource** variable bindings that are sent in the trap identify the trap source. Each of these variable bindings is from the CNAlarmsMib (CheetahNet Alarms MIB)**.** If an EventModel representing the trap source does not exist, it is automatically created.

When the EventModel receives the event, it is processed and can be used to create or clear an alarm. The following table describes how each trap is mapped to a CA Spectrum event and how the event is processed.

Trap OID	Trap Name	Event Generated	Alarm Generated or Cleared	Alarm Severity
1.3.6.1.4.1.1283.10.6.1	Device added	0x3e00001	NA	NA
1.3.6.1.4.1.1283.10.6.2	Device deleted	0x3e00002	NA	NA
1.3.6.1.4.1.1283.10.6.3	Configuration changed	0x3e00003	0x3e00003	Orange
1.3.6.1.4.1.1283.10.6.4	Clear Alarm	0x3e00004	Clears 0x3e00003, 0x3e00005, 0x3e00006, 0x3e00007, 0x3e00008	NA
1.3.6.1.4.1.1283.10.6.5	Warning alarm	0x3e00005		Yellow
1.3.6.1.4.1.1283.10.6.6	Minor alarm	0x3e00006		Yellow
1.3.6.1.4.1.1283.10.6.7	Major alarm	0x3e00007		Orange
1.3.6.1.4.1.1283.10.6.8	Critical alarm	0x3e00008		Red

Chapter 5: HP BladeSystem c-Class

This section describes CA Spectrum support for monitoring the Hewlett-Packard (HP) BladeSystem c-Class device family.

This section contains the following topics:

Overview

Support for the HP BladeSystem c-Class device family is available in CA Spectrum with an enhanced certification. The top-level management uses the model of the HP BladeSystem Onboard Administrator (OA). This device family is modeled and represented in the topology with a OneClick icon representing a chassis.

CA Spectrum chassis device management includes the following features:

- Support for the C7000 and C3000 chassis types.
- OA support, represented in CA Spectrum with a unique model type and chassis icon.
- Automatic blade modeling. After OA modeling, a nontopological module model is created for each occupied chassis slot. These models represent the hardware level view of an occupied slot.
- Automatic chassis identification of previously modeled device models, or managed devices, either pingable or SNMP capable, that are running on a blade.
- Enhanced Interface tab to show the hierarchy view of blades and interfaces for a given chassis. Chassis, managed devices, module models, and interfaces each have a unique icon in the hierarchy.

- Managed devices can be manually associated (or disassociated) with their chassis using a right-click menu option.
- Jump-to navigation from a managed device model to its chassis.
- Jump-to navigation from a module model to its managed device if it exists.
- Support of several chassis-based OneClick views.
- Chassis-based Locator searches.
- Enhanced Fault Isolation capabilities ensure that a single alarm is generated on a chassis-wide failure, eliminating a multiple alarm scenario.

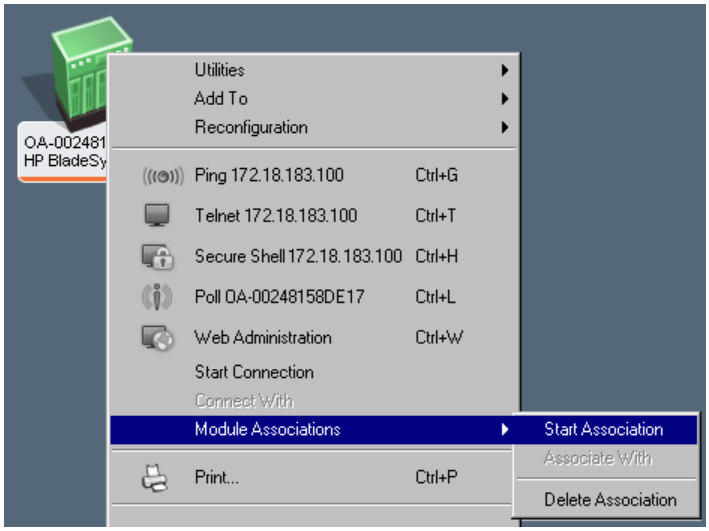
Configuration

Analysis of the chassis modeling environment occurs, by default, every 5 minutes. You can change the polling discovery interval for server and interconnect blades by modifying the *configInterval* attribute. This attribute is located on separate application models that are associated with the HPBladeOnboardAdmin model. For server blades, the relevant application model is HPServerBladeApp. For interconnect blades, the relevant application model is HPNetworkBladeApp.

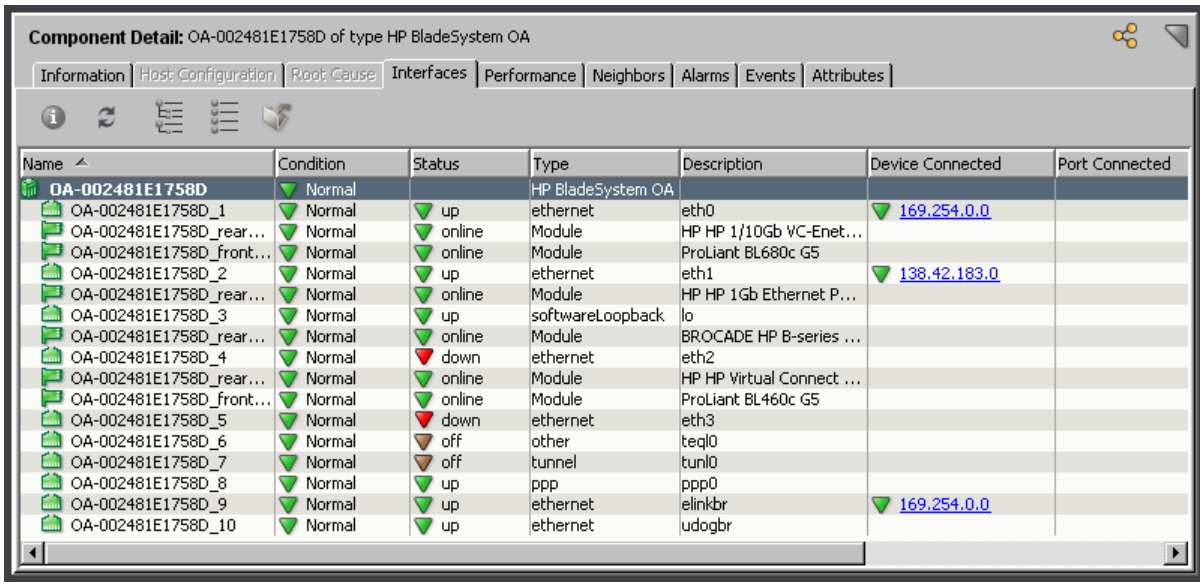
Use the 'By Device IP Address' Locater tab search under Application Models to locate and select the relevant application model. You can modify the *configInterval* attribute using the Attributes tab in the Component Detail panel of OneClick.

Managing Module Associations

Modeling the OA initiates automated module modeling and creates associations between the modules and the chassis. Existing managed device models that can be identified through a serial number are automatically associated with the chassis. HP Insight Manager Agents do provide the required serial number; they are the recommended configuration. Otherwise, use manual association through the 'Module Associations' menu option. Subsequent Module Association menu options let you manage your association with options such as 'Start Association', 'Associate With', or 'Delete Association'.

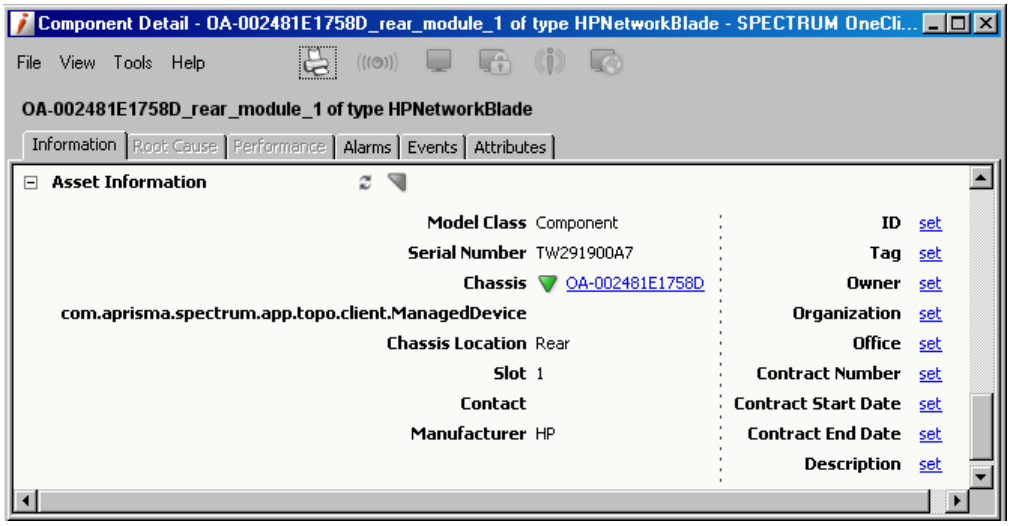


You can view the contained modules and their associated interfaces through the OA Interfaces tab. Supported columns provide the chassis location (front or rear), slot number, module type, and description. The module icon helps you identify the type of hardware.



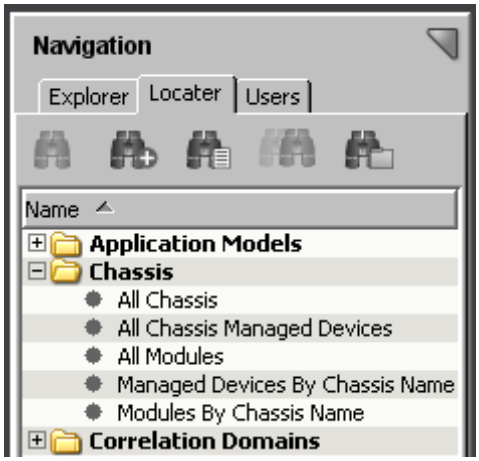
Name	Condition	Status	Type	Description	Device Connected	Port Connected
OA-002481E1758D	Normal		HP BladeSystem OA			
OA-002481E1758D_1	Normal	up	ethernet	eth0	169.254.0.0	
OA-002481E1758D_rear...	Normal	online	Module	HP HP 1/10Gb VC-Enet...		
OA-002481E1758D_front...	Normal	online	Module	ProLiant BL680c G5		
OA-002481E1758D_2	Normal	up	ethernet	eth1	138.42.183.0	
OA-002481E1758D_rear...	Normal	online	Module	HP HP 1Gb Ethernet P...		
OA-002481E1758D_3	Normal	up	softwareLoopback	lo		
OA-002481E1758D_rear...	Normal	online	Module	BROCADE HP B-series ...		
OA-002481E1758D_4	Normal	down	ethernet	eth2		
OA-002481E1758D_rear...	Normal	online	Module	HP HP Virtual Connect ...		
OA-002481E1758D_front...	Normal	online	Module	ProLiant BL460c G5		
OA-002481E1758D_5	Normal	down	ethernet	eth3		
OA-002481E1758D_6	Normal	off	other	teql0		
OA-002481E1758D_7	Normal	off	tunnel	tunl0		
OA-002481E1758D_8	Normal	up	ppp	ppp0		
OA-002481E1758D_9	Normal	up	ethernet	elinkbr	169.254.0.0	
OA-002481E1758D_10	Normal	up	ethernet	udogbr		

From the perspective of a module model, you can identify the parent chassis using the Chassis navigation link in the Asset Information OneClick view. You can also identify an associated managed device, if one exists, using the Managed Device link in the same view.



Locating Chassis

From the Locater tab of the Navigation panel, you can now see the following Chassis search menu options. This feature assists you in changing the polling discovery interval.



All Chassis

Displays all chassis models (HP OA model, for example)

All Chassis Managed Devices

Displays all device models that are managed through CA Spectrum that are running on a blade. This search only includes pingable or SNMP-capable device models. The module models that are created for every occupied slot in a chassis are not included.

All Modules

Displays all module models, one for every occupied slot in a chassis. The search does not include the managed devices (SNMP- or ICMP-capable). They represent the hardware-level view of an occupied slot.

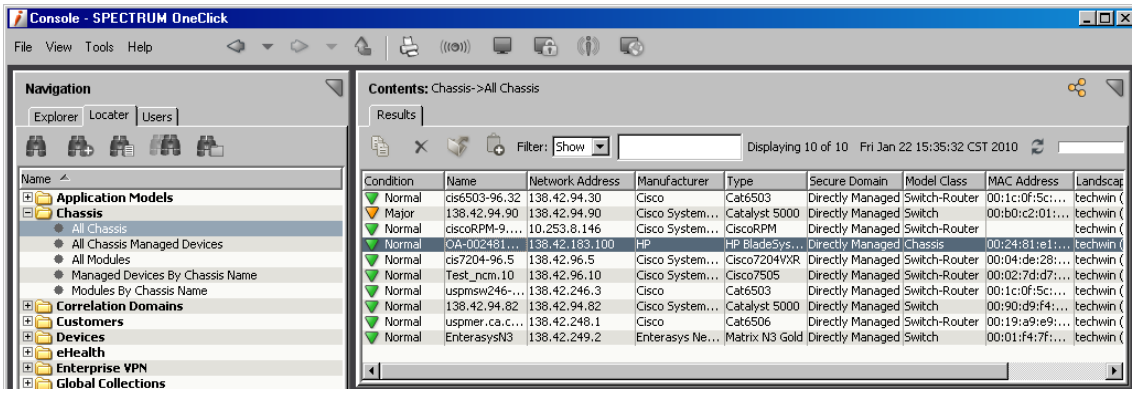
Managed Devices by Chassis Name

Displays all device models that CA Spectrum manages and that are running on a blade on a specified chassis. A subsequent window lets you enter the specific chassis name whose associated devices you want to view.

Modules by Chassis Name

Displays all module models for the specified chassis. A subsequent window lets you enter the specific chassis name whose associated modules you want to view.

As an example, select the 'All Chassis' chassis search option. The following results appear in the Contents panel:



Chapter 6: Juniper M Series

This section describes the Redundant Component Monitoring intelligence available for support of JnprRedundRtr (M20, M40e, and M160) routers in CA Spectrum.

This section contains the following topics:

Redundant Component Monitoring Intelligence

Juniper M20, M40e, and M160 routers support the Redundant Component Architecture. Redundant components include those pieces of hardware that are necessary for proper routing functionality. The following specific components for these routers are passive monitoring and active monitoring.

Note: All Juniper M Series routers can be modeled, without this functionality, by type as a JNPR_Mxxx. This feature provides basic modeling functionality as is described for the JNPR_Mxxx model type.

When either Passive Monitoring or Active Monitoring is invoked, they check for status changes in each type of redundant component. The redundant components of Juniper M Series routers differ based on the following router models:

- **Juniper M20 -** System and Switch Board(s), Routing Engine(s).
- **Juniper M40e** - Routing Engine(s), Miscellaneous Control System(s), System and Forwarding Module(s), PFE Clock Generators.
- **Juniper M160** - Routing Engine(s), Miscellaneous Control System(s), System and Forwarding Module(s), PFE Clock Generators.

Passive Monitoring

Passive Monitoring intelligence reports changes in the status of redundant components only after CA Spectrum has lost contact with the router. When contact is reestablished with the device, CA Spectrum queries the device. The query determines whether component status changes have occurred. Passive Monitoring is always on, but it only checks for component status changes in the previously mentioned case.

Note: When contact is reestablished with the device, the components are not always in a “steady” state. The components take a few minutes to reach their steady state. Each router type (M20, M40e, or M160) takes a different amount of time to reach its steady state. As a result, Passive Monitoring waits 60, 90, or 120 seconds before checking the states of the M20, M40e, or M160 components.

Active Monitoring

Active monitoring is used to report changes in the status of redundant components. The value of the Active Polling Interval determines the frequency of active monitoring. This interval determines how often (in seconds) the Active Monitoring intelligence queries the device to find component status changes. The field is read/write. For example, if the Active Polling Interval is set to 60, the device is queried every 60 seconds for component status changes. When Active Monitoring is enabled, it works in addition to the functionality provided by Passive Monitoring.

You have several other options for enabling or disabling this functionality. First, you can set the Active Polling Interval to 0 to disable active polling. Change the value to a value in seconds to enable this functionality when the Active Monitoring attribute is set to True. Changing the Polling Status of the device model to False also disables Active Monitoring.

Setting the Polling Interval of the device model to 0 also disables Active Monitoring. Changing Polling Status to True or changing the Polling Interval to a nonzero value does not reenable Active Monitoring, when Active Monitoring is disabled.

Never set the Active Polling Interval to a value that is less than the "steady" state time for the given router type. For example, the "steady" state time for an M20 is 60 seconds. Set the Active Polling Interval to a value that is greater than 60.

The following attributes control the Active Monitoring intelligence:

- **ActiveMonitor** - Enables or disables the active monitoring intelligence. The default value is disabled.
- **ActivePollInt** - Determines the frequency (in seconds) of Active Monitoring queries to the device for component status changes.

Chapter 7: Netscreen Firewall

This section describes the Netscreen Tunnel Interface model type (nsTunnelIf) and its functionality.

This section contains the following topics:

Tunnel Interfaces

This section describes CA Spectrum support for monitoring NetScreen Firewall tunnel interfaces.

Model Tunnel Interfaces

Various attributes control whether the site-to-site Tunnel Interfaces are modeled on your Netscreen device. You can model other types of tunnel interfaces by using the following procedure. By default CA Spectrum does not model Dialup Tunnels or Tunnels whose monitor state is set to OFF. To enable the modeling of these types of tunnels, use the Model Type Editor.

Follow these steps:

1. Shut down the SpectroSERVER and start the Model Type Editor.
2. To enable modeling of Dialup Tunnels, use the Search text box on the Attributes tab to find the TunnelFilterTypes attribute (0x12a17) of the NSFirewallVPN model type.
3. Remove the value 1 from the list of values for this attribute.
4. To enable modeling of tunnels whose monitor state is OFF, use the Search text box on the Attributes tab to find the TunnelFilterStates attribute (0x12a19) of the NSFirewallVPN model type.
5. Remove the value 0 from the list of values for this attribute.
6. Save your changes in the Model Type Editor, and restart the SpectroSERVER.
7. Reconfigure the Netscreen models using the Manually Poll Device option that is available for each device model.

   The tunnel interfaces are modeled.

Tunnel Interface “Stacking”

Tunnel interface models are created as subinterfaces of the physical interface whose IP address matches the local address of the tunnel. This behavior is indicated in the VPN-MON.mib. Because NetScreen devices do not support the ifStackTable, this mechanism for determining the lower-layer interface is necessary and effective.

Automatic Connectivity Mapping

A tunnel interface model activates for the first time during initial device modeling or during an interface reconfiguration. Then CA Spectrum searches for a tunnel interface model that represents the other end point of the tunnel. If such a model is found, the connection between these two interfaces is modeled. CA Spectrum uses the local address and remote address that are indicated in the VPN-MON.mib to find the other end point of the tunnel.

Interface Model Identification

You can identify a Tunnel interface model by its local address and remote address, as indicated in the VPN-MON.mib. This identification method lets CA Spectrum preserve the interface model if the ifIndex of the interface changes.

Status Monitoring of Tunnel Interfaces

On the NetScreen device, the ifOperStatus of a tunnel interface entry is always "UP until it disappears from the ifTable. If a tunnel model becomes "stale", and no link down trap is processed for the tunnel, CA Spectrum generates a red alarm on the model.

This alarm is suppressed in the following cases:

- If the physical interface is down (the same case in which a link down trap alarm is suppressed).
- If the "Suppress Linked Port Alarms" setting of the Live Pipes model is set to True, and either of the following conditions are met:
  - The connected device is unreachable (by the SpectroSERVER)
  - The "linked" tunnel interface model has an alarm (red)

This status monitoring functionality is only available when Live Links are enabled for the port that is associated with the tunnel interface. For information about enabling Live Links, see the *Modeling and Managing Your IT Infrastructure Administrator Guide*.

CA Spectrum Management Settings

The following CA Spectrum management settings are recommended.

Automatically Reconfigure Interfaces

Set this attribute to True for NetScreen models if you want CA Spectrum to manage the branch tunnels of the device. For devices that only support "User" tunnels, set this attribute to False. When set to True, CA Spectrum reconfigures the interface models whenever the ifNumber object of an SNMP agent of the device changes.

Reconfigure on LINK change

We recommend setting the value of this attribute to False for all NetScreen models. When set to True, CA Spectrum performs an interface reconfiguration after every 'link up' or 'link down' trap that it receives.

Discovery after Reconfigure Attribute

We recommend retaining the default value of False for the Discovery after Reconfigure attribute for all NetScreen models. CA Spectrum models connections between newly found tunnels regardless of this setting. The CA Spectrum autodiscovery process adds little or no value after most link state changes, especially for NetScreen devices. For these devices, most link state changes represent tunnels coming up and going down, and not the configuration of new router or bridge ports.

Create Sub-Interfaces

Set this attribute to True for NetScreen models if you want CA Spectrum to monitor the branch tunnels. If this attribute is set to False, CA Spectrum does not create models for the tunnel interfaces.

Suppress Linked Port Alarms

We recommend setting this attribute of the Live Pipes model to True. This setting suppresses port alarms when the connected device is unreachable or the linked port model already has an alarm.

Chapter 8: Nortel Contivity VPN Switches

This section describes CA Spectrum support for monitoring Nortel Contivity VPN switches.

This section contains the following topics:

Tunnel Interfaces

This section describes the Tunnel Interface Filter functionality for Nortel Contivity devices.

Tunnel Interface Filtering

The ContivityVPN device populates the **ifTable** with both user and branch VPN tunnel interface entries. However, thousands of user VPN tunnel interfaces can exist. The ContivityVPN interface filtering functionality filters out user tunnel interfaces and prevents unnecessary modeling of these interfaces.

Note: Tunnel interface filtering is only available for models of type **ContivityVPN**.

Enabling and Disabling Tunnel IF Filtering

The following steps enable or disable Tunnel IF filtering:

Follow these steps:

1. In the Model Type Editor, set the default list value for the attribute If_Mtype_Map handle 0x011fb4.
2. Look at the list of values, and locate OID instance 131.
3. Set to a value of 0. This setting prevents the interface type from being modeled.
4. To disable the tunnel interface filtering and enable model creation, set this value to 220013.

Modeling of Tunnel Interfaces

The Create Sub-Interface attribute of the Contivity device model controls the creation of models to represent site-to-site or branch tunnel interfaces. No models are ever created to represent "user" tunnels. This behavior is consistent with previous versions.

Tunnel Interface "Stacking"

Tunnel interface models are created as sub-interfaces of the physical interface. The IP address of the physical interface matches the local address of the tunnel as indicated in the Tunnel MIB. The Contivity devices do not support the ifStackTable. As a result, this mechanism of determining the lower-layer interface is necessary and effective.

Automatic Connectivity Mapping

A tunnel interface model activates for the first time during initial device modeling or during an interface reconfiguration. Then CA Spectrum searches for a tunnel interface model that represents the other end point of the tunnel. If such a model is found, the connection between these two interfaces is modeled. CA Spectrum uses the local address and remote address that are indicated in the Tunnel MIB (rfc2667) to find the other tunnel end point.

Interface Model Identification

You can identify the Tunnel interface models by their local address and remote address, as indicated in the Tunnel MIB (rfc2667). This identification lets CA Spectrum preserve the interface model even if the ifIndex of the interface changes.

Interface Model Aging

During an interface reconfiguration, any interface model that is no longer represented in the MIB is marked as "stale" instead of being destroyed. This feature lets CA Spectrum retain the connectivity modeling between tunnel interfaces and other devices while the tunnel is down. The connectivity information can then be used for the event correlation and fault suppression.

On subsequent reconfigurations, the port age-out time of the device model is compared with the time period that the interface model has been stale. If the interface does not reappear in the MIB, the interface model is destroyed after it ages out. If the interface does reappear in the MIB, the interface model is marked as "current." The port is marked as stale by setting the "isStale" attribute to True. You can set the port age-out time per device Set the "PortAgeOutTime" on the device to a number of minutes. The default age-out time for the Contivity device is two hours (120 minutes).

Link Down Trap Correlation

To avoid sending multiple alarms for a single network outage, link down traps for "tunnel" interface models are correlated with other conditions. The alarms for link down traps are suppressed when the lower layer (that is, the physical interface) is down. When the "Suppress Linked Port Alarms" setting of the Live Pipes model is set to True, the alarms for the link down traps are suppressed. The alarms are suppressed under the following conditions:

1. The connected device is unreachable (by the SpectroSERVER).
2. The "linked" tunnel interface model has an alarm (is red).

Status Monitoring of Tunnel Interfaces

On the Contivity device, the ifOperStatus of a tunnel interface entry is always "UP" until it disappears from the ifTable. When a tunnel model becomes "stale" and link down traps have not been processed for the tunnel, CA Spectrum generates a red alarm on the model. The red alarm is suppressed in the same cases in which a link down trap alarm is suppressed. The red alarm is suppressed when the lower layer (that is, physical interface) is down. When the "Suppress Linked Port Alarms" parameter of the Live Pipes model is set to True, this alarm is suppressed.

The alarm is suppressed under the following conditions:

1. The connected device is unreachable (by the SpectroSERVER).
2. The "linked" tunnel interface model has an alarm (red).

Contivity Management Settings

The following Contivity settings are recommended.

Enable Tunnel MIB

We recommend enabling the Tunnel IP MIB on all managed Contivity devices. This setting lets CA Spectrum create models to represent the tunnel end points on the device. This MIB can be enabled and disabled from the ADMIN->SNMP section of the Contivity web management pages.

Enable Link Up/Down Traps

We recommend enabling link up and link down traps for physical interfaces and for "Nailed-Up" branch tunnels. This setting gives CA Spectrum immediate notifications of link state changes. Our testing has shown that link traps for "OnDemand" tunnels do not provide much value. The tunnel must be down for approximately 15 minutes before the trap is sent.

Nail-Up Your Monitored Tunnels

We recommend that all tunnels for which connection monitoring is important be "Nailed-Up". CA Spectrum does not alarm "OnDemand" tunnels when they go down. Specifically, the Alarm on LINK down Trap attribute of the Tunnel_If model determines whether it responds to link down traps or changes to the isStale attribute. A value of Always (1) causes CA Spectrum to process these events; a value of Never (0) causes CA Spectrum to ignore them. When CA Spectrum creates the Tunnel_If models for the Contivity, it sets this attribute to Always for "Nailed-Up" branch tunnels, and Never for "OnDemand" tunnels.

Change the Alarm on LINK down setting from the Configuration tab of the Global Attribute Editor. We recommend leaving it as CA Spectrum has set it.

CA Spectrum Management Settings

The following CA Spectrum management settings are recommended.

Automatically Reconfigure Interfaces

Set this attribute to True for the Contivity models if you want CA Spectrum to manage the branch tunnels of the device. For the devices that only support "User" tunnels, set this attribute to False. When set to True, CA Spectrum reconfigures the interface models whenever the ifNumber object of the SNMP agent changes on the device.

Reconfigure on LINK change

We recommend setting this attribute to False for all Contivity models. When it is set to True, CA Spectrum performs an interface reconfiguration after every link up or every link down trap is received.

Discovery after Reconfigure

We recommend retaining the default value of False for the Discovery after Reconfigure attribute for all Contivity models. CA Spectrum models connections between newly found tunnels regardless of this setting. The CA Spectrum autodiscovery process adds little or no value after most link state changes, especially for Contivity devices. For these devices, most link state changes represent tunnels coming up and going down, and not the configuration of new router or bridge ports.

Create Sub-Interfaces

Set this attribute to True for Contivity models if you want CA Spectrum to monitor the branch tunnels. If this attribute is set to False, CA Spectrum does not create models for the tunnel interfaces.

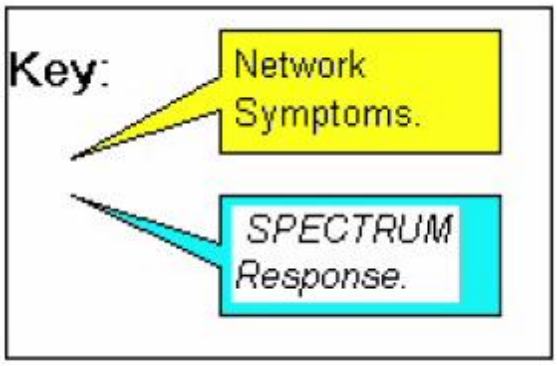
Suppress Linked Port Alarms

We recommend setting this attribute of the Live Pipes model to True. This setting suppresses port alarms when either the connected device is unreachable or the linked port model already has an alarm.

Contivity Fault Scenarios

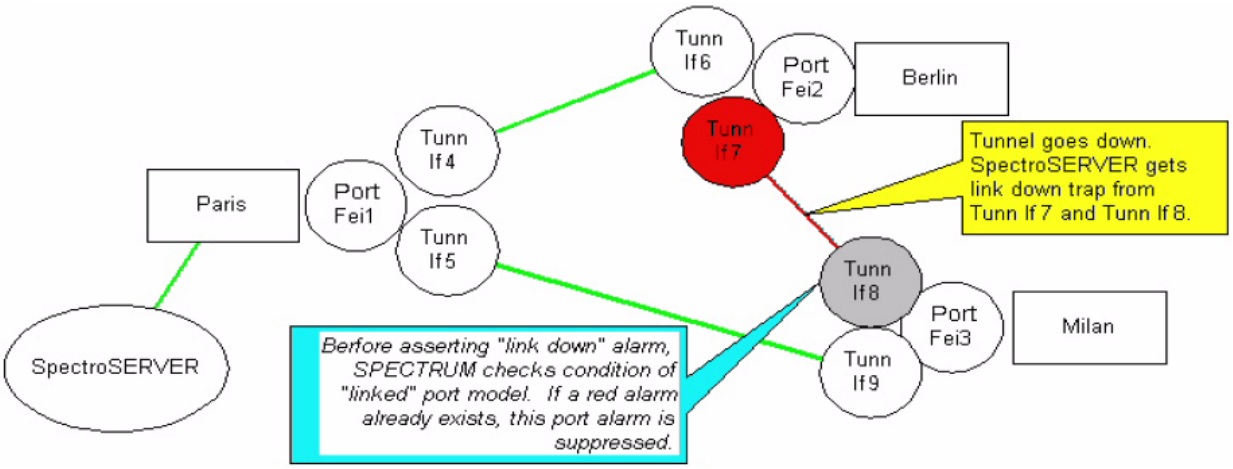
This section describes fault scenarios that are likely in a VPN environment and the CA Spectrum response to each scenario.

The following key applies to each of the diagrams in this section:



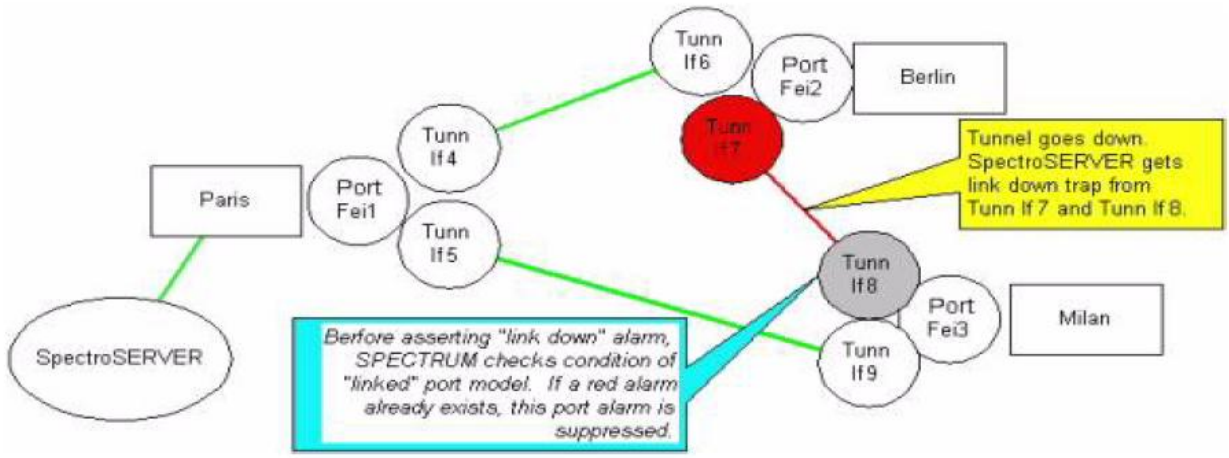
Two Link Down Traps for One Down Tunnel

In the following scenario, the SpectroSERVER retains contact to all managed elements in this meshed environment, but a tunnel between two devices goes down. CA Spectrum receives two link down traps. One tunnel interface alarms; the other alarm is suppressed.



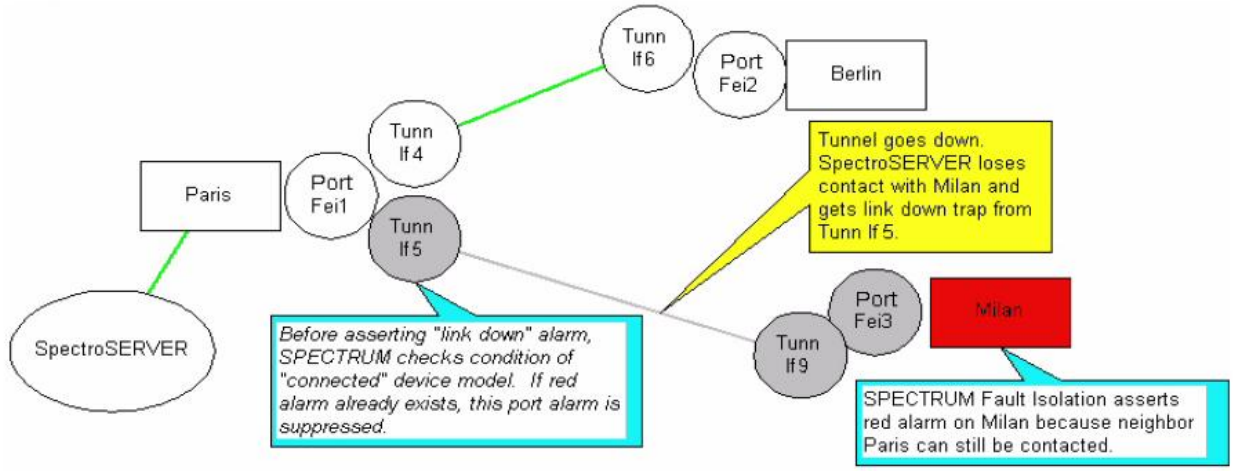
Loss of Contact and Link Down Trap

In the following scenario, CA Spectrum loses contact with a "spoke" Contivity in a hub and spoke network. CA Spectrum also receives a link down trap from the hub, indicating the tunnel to the lost device. CA Spectrum sends an alarm for the lost device and suppresses the alarm on the tunnel interface that is indicated by the trap.



Physical Port Down, Loss of Contact, and Link Down Traps

In the following scenario, a physical port of a Contivity goes down or loses its link to the public network. CA Spectrum gets link down traps for the physical port and tunnels of the Contivity, and loses contact with remote the Contivity devices. The link down alarms on the tunnel interface models are suppressed, but CA Spectrum fault isolation creates red alarms on the lost Contivity device models because they have an "up" neighbor.



Known Anomalies

CA Spectrum contains the following known anomalies.

Sub-Interface Changes

When Create Sub-Interfaces is changed from True to False for a Contivity model after tunnel interface models have been created, the tunnel interface models are not destroyed immediately after an interface reconfiguration. Instead, these models go stale and start aging out. To enable tunnel monitoring for a subset of Contivity devices, set the default value of Create Sub-Interfaces to False. Then set Create Sub-Interfaces to True for the individual models of Contivity devices that require tunnel monitoring.

Autodiscovery and Public Addresses

Generally, the public addresses on the Contivity devices in a VPN are in different subnets because multiple routers separate them. The Contivity devices with public interfaces can be on the same subnet. In this case, CA Spectrum autodiscovery can attempt to map the connectivity of the public interfaces. The result would be a LAN container in the same topology view as the Contivity models with pipes to the Contivity models. A fanout model without the LAN would be connected to the public interface models of the Contivity devices.

Port Aging

CA Spectrum port aging is not aggressive. When a tunnel becomes inactive, the tunnel interface model is marked as "Stale". Any future reconfiguration that occurs after the "portAgeOutTime" of the device causes that tunnel model to be destroyed. However, if no future reconfigurations of the device occur, the "Stale" tunnel interface model remains.

For example, consider a polling interval of 5 minutes and a portAgeOutTime of 30 minutes. If a tunnel goes down at 10:27 and CA Spectrum polls at 10:30, CA Spectrum detects an ifNumber change and performs and interface reconfiguration. During this process, the tunnel interface is marked as stale. If the tunnel does not come back up, the tunnel interface model is destroyed at 11:00. When ifNumber does not change again for a week, interface reconfiguration cannot run again for a week. This tunnel interface model remains stale for one week and is then destroyed.

Index