HP Network Node Manager i Software
Step-by-Step Guide to Deploying NNMi

NNMi 9.20

This document describes deploying a new NNMi 9.20 installation on a small test network. The steps included are similar to those you would take to deploy NNMi in a production network.

Read this document, and then use the *HP Network Node Manager i Software Deployment Reference* as a resource. It contains many details that extend beyond the technical scope of this document.

To find the latest *HP Network Node Manager i Software Deployment Reference, see* http://h20230.www2.hp.com/selfsolve/manuals.

CONTENTS

The Basic Steps: A Roadmap

This document assumes you have completed the following prerequisites:

- You have installed NNMi.
- Your server meets all the system prerequisites, including the patch requirements and kernel parameters shown in the *HP Network Node Manager i Software System and Device Support Matrix*, available at http://h20230.www2.hp.com/selfsolve/manuals.

Caution: The NNMi installation script does not check that your server meets the system prerequisites. Ignoring these requirements can cause issues after you complete your installation.

The examples in this document are of an NNMi installation on a Linux server. If you are using NNMi installed on a Windows server, convert any paths and commands.

This document describes the following tasks:

1. Apply the License
2. Back up the Original Configuration
3. Sign in to NNMi and Create Users
4. Set up Communication Configuration
5. Configure Discovery
6. Configure Monitoring
7. Configure Incidents, Traps, and Automatic Actions
8. Configure the NNMi Console
9. Maintain NNMi
10. Check NNMi Health

It also includes Best Practices and Example Usage Scenarios.

See the *HP Network Node Manager i Software Deployment Reference*, available at http://h20230.www2.hp.com/selfsolve/manuals, for information about the following topics:

1. Security Groups and Multi-tenancy
2. Integration with other HP products such as HP Operations Manager (HP OM), HP Universal Configuration Management Database (HP UCMDB), and third-party products
3. High Availability or Application Failover
4. Using a remote Oracle database
5. NNM iSPIs, such as NNM iSPI for Performance and NNM iSPI for MPLS

Apply the License

You can use the instant-on license or obtain a larger temporary license from HP.

Contact your HP Sales Representative or your Authorized Hewlett-Packard Reseller for information about the NNMi licensing structure, and to learn how to add license tiers for enterprise installations. To obtain additional license keys, go to the HP License Key Delivery Service: https://webware.hp.com/welcome.asp

Note: The instant-on license enables NNMi for 250 nodes.

You can install the license using the command line. The following command shows an example of installing the license using the `nnmlicense.ovpl` script:

```
nnmlicense.ovpl NNM -f ./mylicense.key
```

Back up the Original Configuration

Make a backup of the original NNMi configuration before making any changes. This way, you can revert back to the original configuration if needed.

To back up the original NNMi configuration, complete the following steps:

1. Create a directory on the NNMi management server where you want to keep the original configuration files. For this example, create a directory called /var/tmp/origconfig.

2. Run the **nnmconfigexport.ovpl** command using the **-c** and **-f** options. The **−c** option specifies *all configurations* and the **−f** option specifies the directory.

   The following command shows an example of running the `nnmconfigexport.ovpl` script:

   nnmconfigexport.ovpl -c all -f /var/tmp/origconfig/

   After you run the **nnmconfigexport.ovpl** script, NNMi displays output similar to the following:

   ```
   Successfully exported /var/tmp/origconfig/incident.xml.
   Successfully exported /var/tmp/origconfig/status.xml.
   …
   Successfully exported /var/tmp/origconfig/account.xml.
   Successfully exported /var/tmp/origconfig/securitymappings.xml.
   Successfully exported /var/tmp/origconfig/security.xml.
   ```
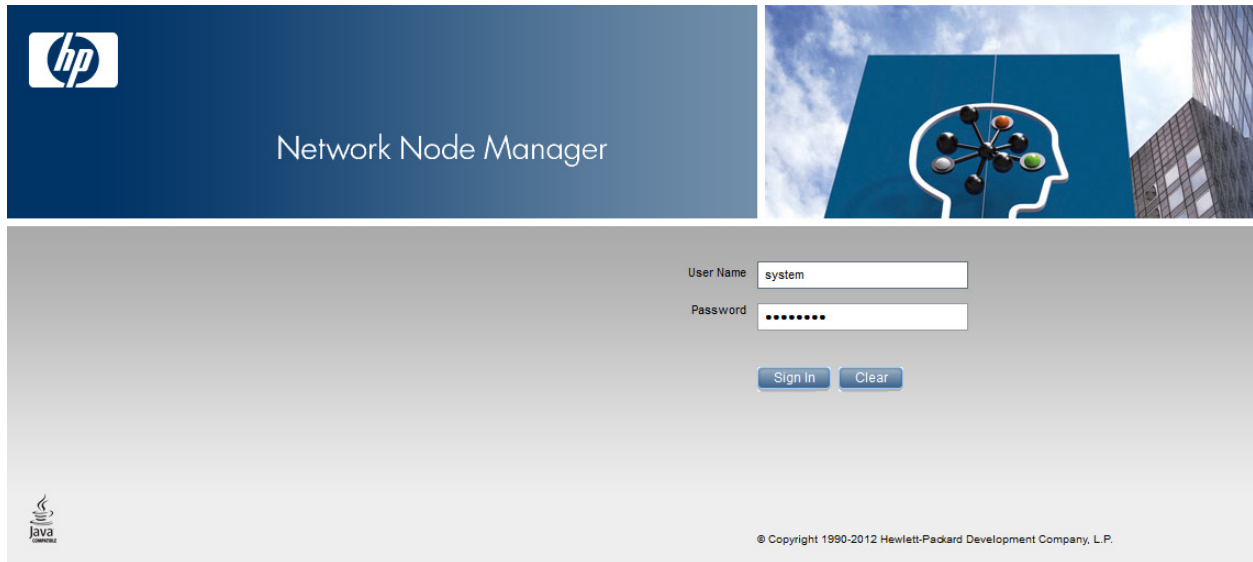
Sign in to NNMi and Create Users

Initial Sign In

Access NNMi using a browser such as Internet Explorer or Mozilla Firefox. Use a URL similar to the following, inserting your server name and the port you selected for communication during the installation process:

```
http://<serverName>:<port number>/nnm
```

Figure 1: NNMi Sign In Screen

Create User Accounts and Roles

Do not use the `system` user name in most cases. Create and use an Administrator account for most of your work, following these instructions:

1. From the workspace navigation panel, select the **Configuration** workspace.
2. Expand the **Security** folder.
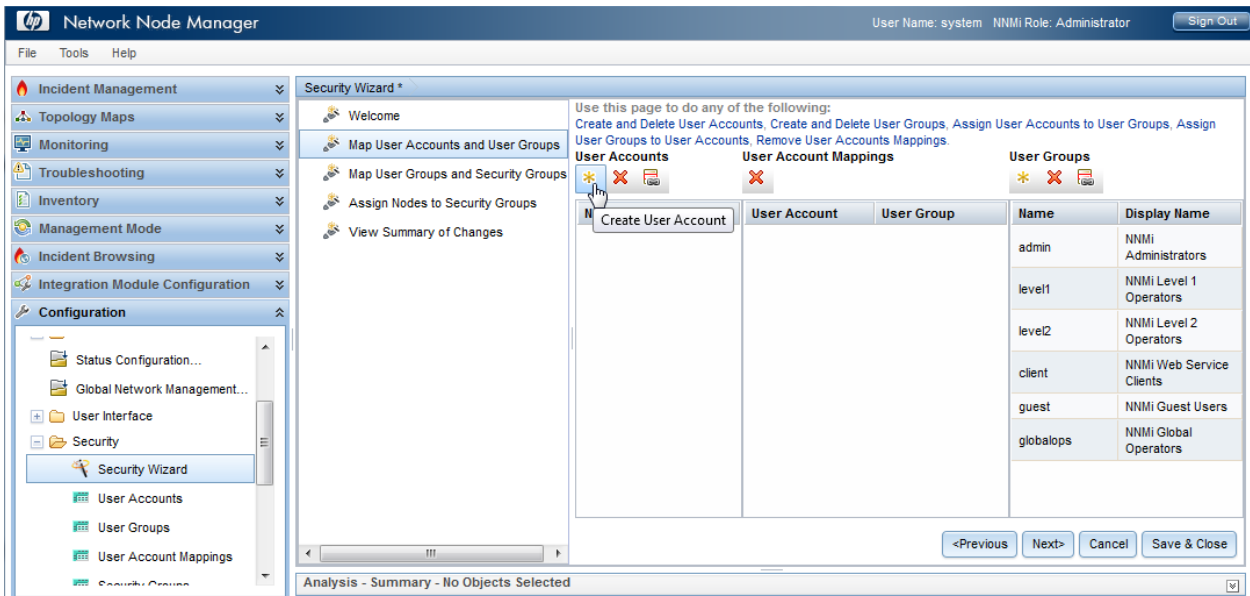3. Click **Security Wizard**, and then click **Next**.

You should see the Security Wizard Welcome Page.
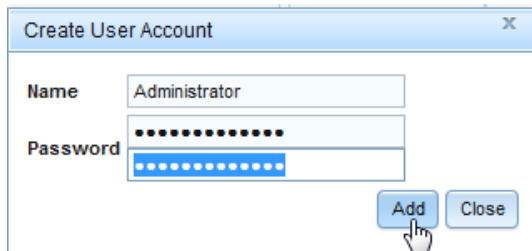
Figure 2: Security Wizard: Welcome Page
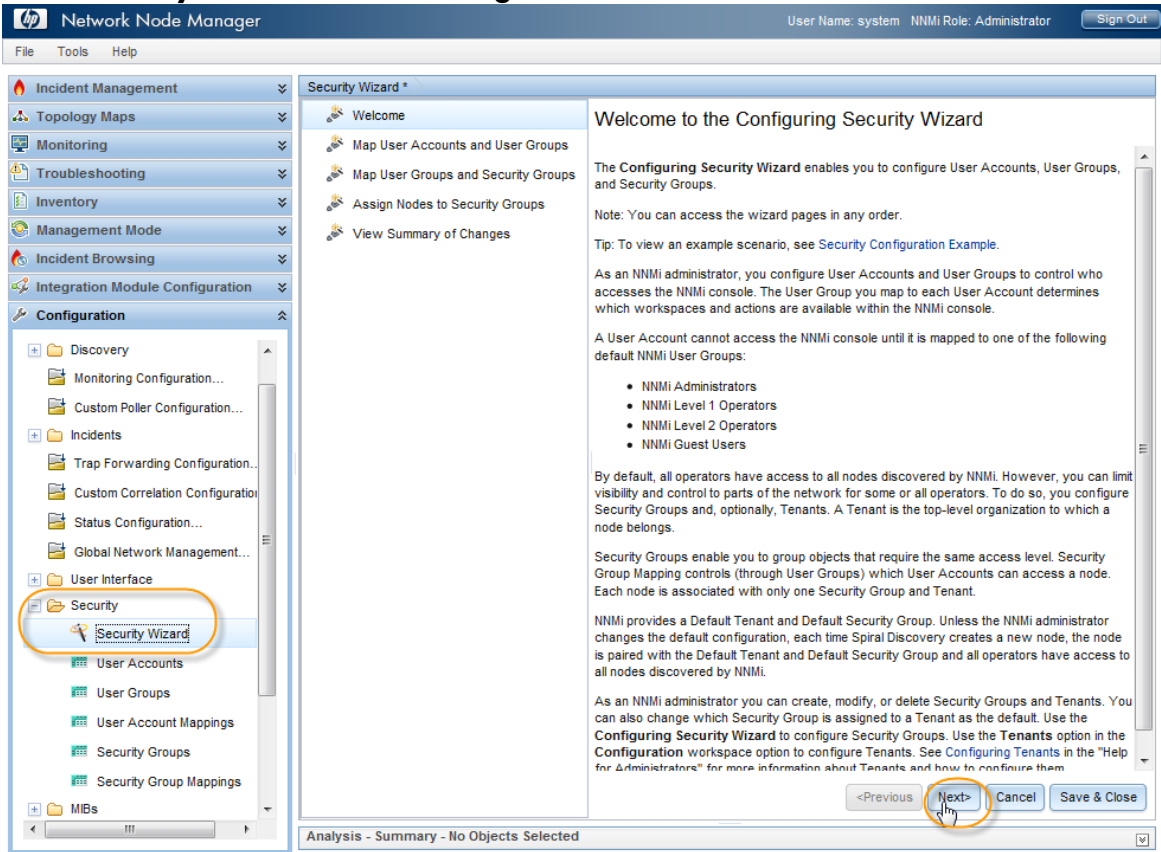
4. Navigate to **User Accounts** and click the ✱ icon.

Figure 3: Security Wizard: Create User Account

5. In the Create User Account dialog box, enter the account information, click **Add**, and then click **Close**.

Figure 4: Security Wizard: Create User Account Dialog Box

6. Click the new account name in the **User Accounts** column, and then click the ◁ icon next to the appropriate User Group to create the User Account Mapping.
7. Click **Save and Close**, and then click **OK > OK** to accept the changes.

   Tip: User Account Mappings replace the "Role" concept in previous versions of NNMi.

Figure 5: Security Wizard: Assign User Group to User Account
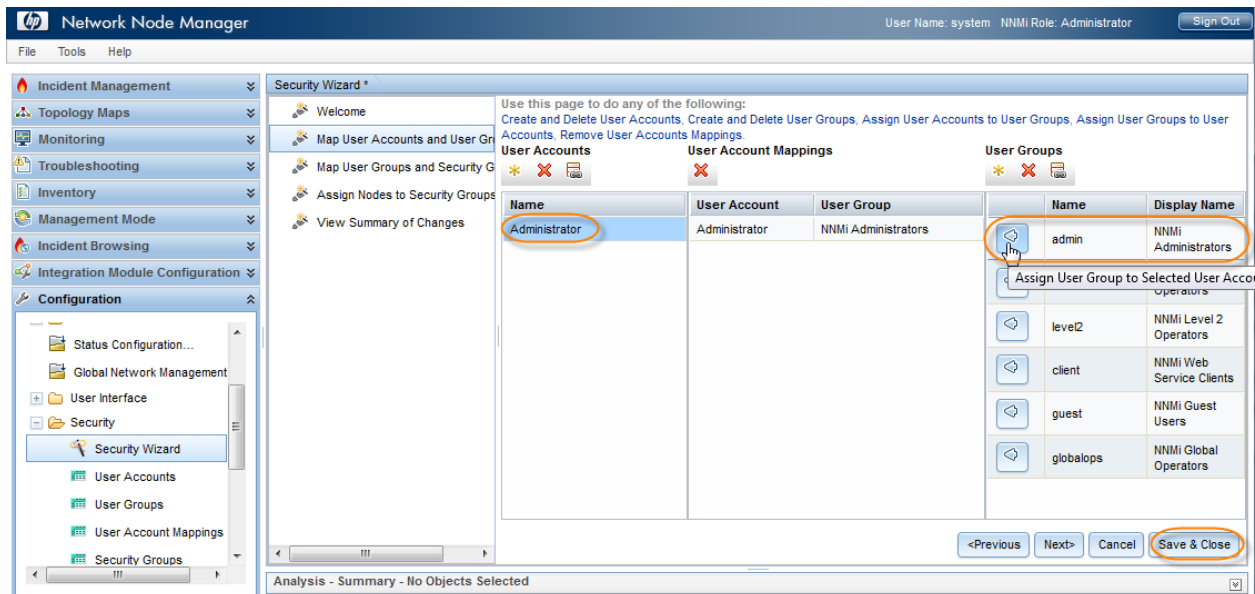
8. Sign out of NNMi and sign in with the new User Account Name to make sure it works correctly.

Set up Communication Configuration

By default, NNMi performs *SNMP community string discovery*. This example describes how to use this default method.

Tip: Unlike previous versions of NNMi, you do not configure a prioritized list of SNMP community strings.

By default, NNMi tries all possible community strings sequentially. NNMi selects the first community string that results in a response from a node as the SNMP community string for that node. In this example, configure only the default community strings. You can implement more complex solutions with this configuration, but in most cases, this is an adequate approach.

1. From the workspace navigation panel, select the **Configuration** workspace, and then click **Communication Configuration**.
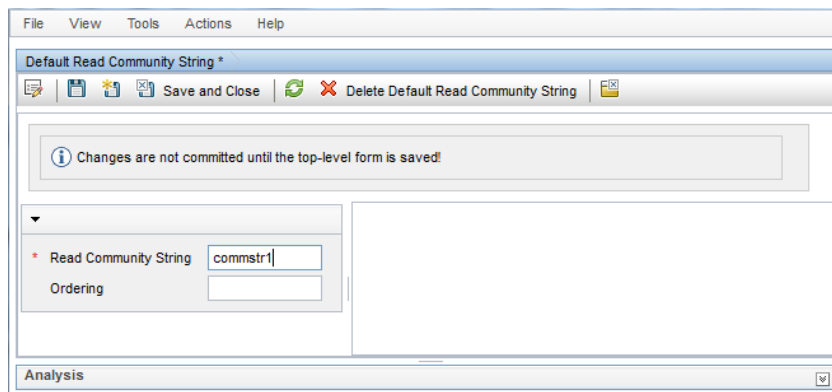
Figure 6: Communication Configuration
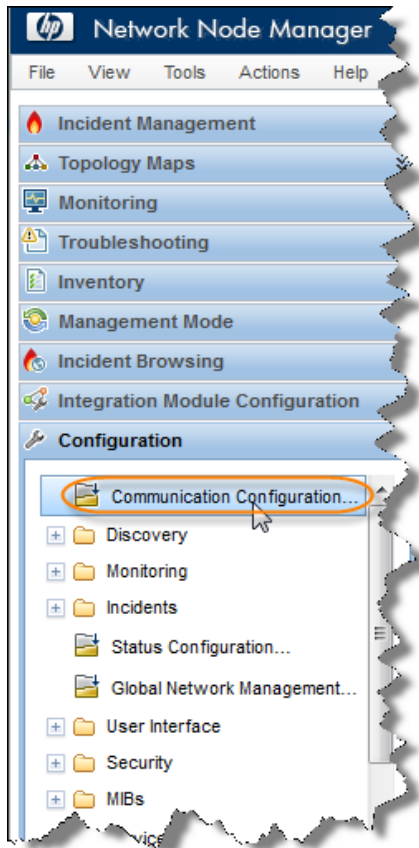
2. Click the **Default SNMPv1/v2 Community Strings** tab, and then click the ✳ icon to create a new community string.

Figure 7: Communication Configuration: Default SNMPv1/v2 Community Strings Tab
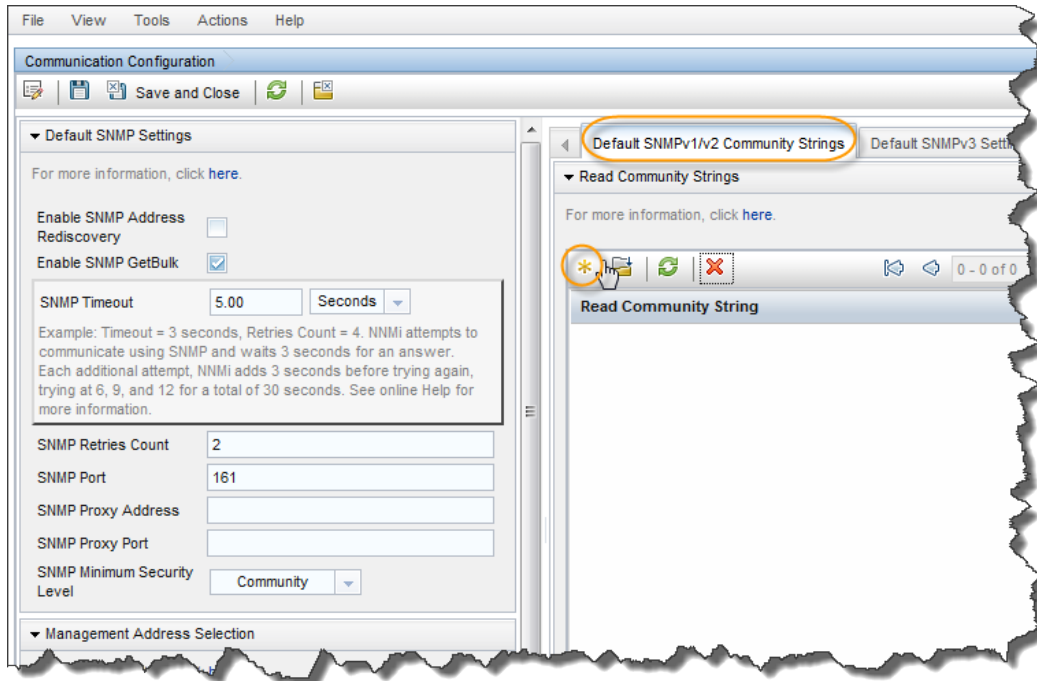
3. Enter your community string, and then click ⊞ **Save and Close**.

Figure 8: Default Read Community String

4. Repeat the previous steps for all your community strings.

Tip: Explore the other Communication configuration options in case you want to make additional changes.

5. When you finish configuring your community strings, click ⊞ **Save and Close** in the **Communication Configuration** form to save your changes.

Your SNMP configuration is complete.

Figure 9: Communication Configuration: Save and Close
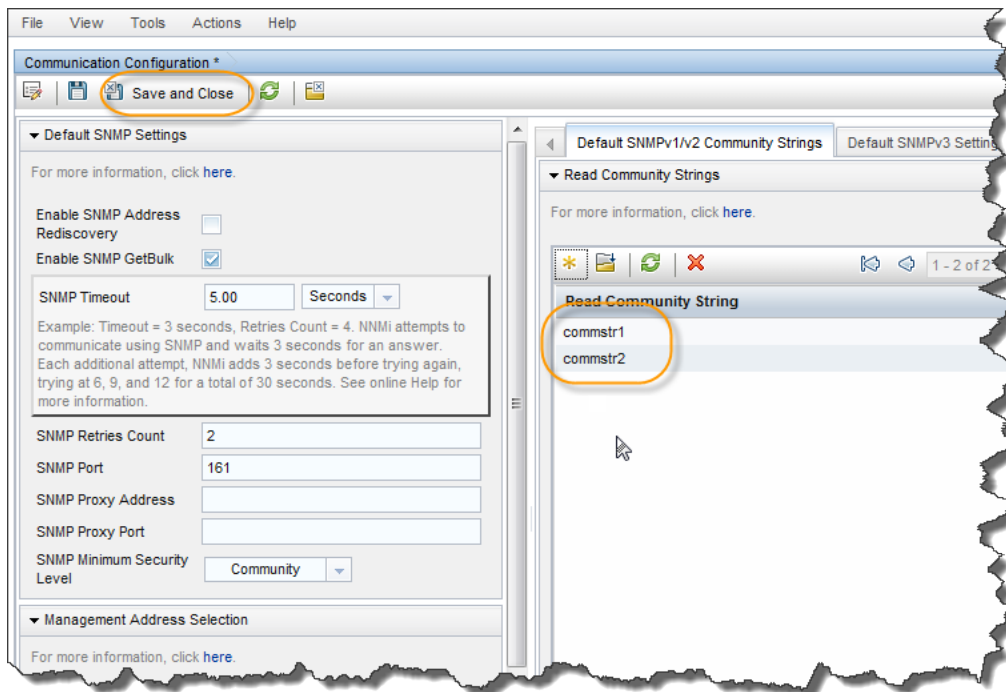
Configure Discovery

NNMi supports two methods of discovery: *list-based and automatic*. Each method offers advantages.

List-based discovery uses a list of node names or addresses as input and only discovers the nodes contained in that list. NNMi discovers no additional node names or addresses beyond those contained in this list. This method gives you control over what is discovered and managed by NNMi. Each node in the list is known as a seed.

Note: NNMi loads each seed even if its IP address is outside of the Auto-Discovery range.

Tip: It you load a seed as an IP address for a device, it is a good practice to specify the preferred management address (usually the loopback address with Cisco gear) as the seed.

Automatic discovery finds nodes on the network based on user-specified criteria. You can configure NNMi to restrict discovered nodes based by address range, SNMP values (such as system object ID), device type, and other methods. You can configure automatic discovery with a single seed node; although even this node is not required if you enable the optional *ping- sweep* feature.

The following example describes an automatic discovery based on an address range. Additionally, this example shows you how to load a couple of seed nodes.
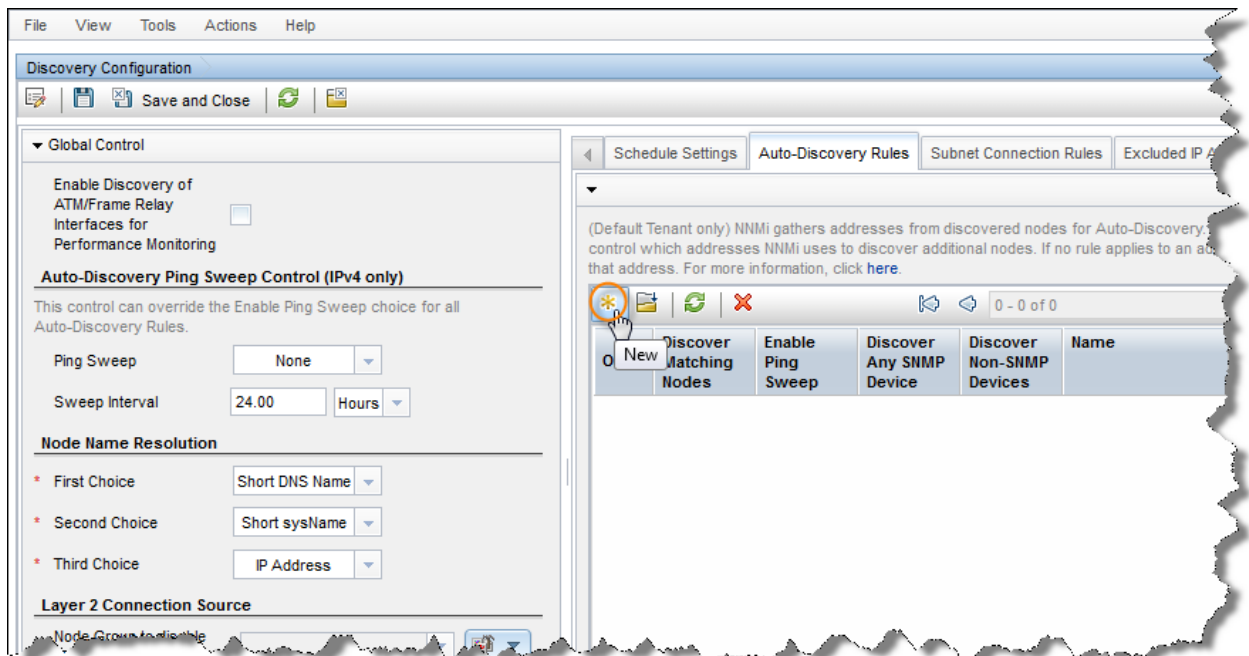
1. From the workspace navigation panel, select the **Configuration** workspace, expand the **Discovery** folder, and then click **Discovery Configuration**.

Figure 10: Discovery Configuration
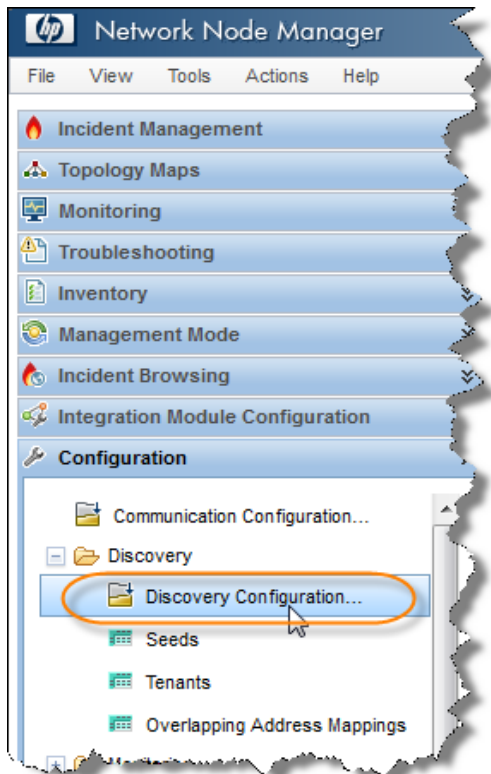
2. Click the **Auto-Discovery Rules** tab, and then click the ✳ icon to create a new rule.
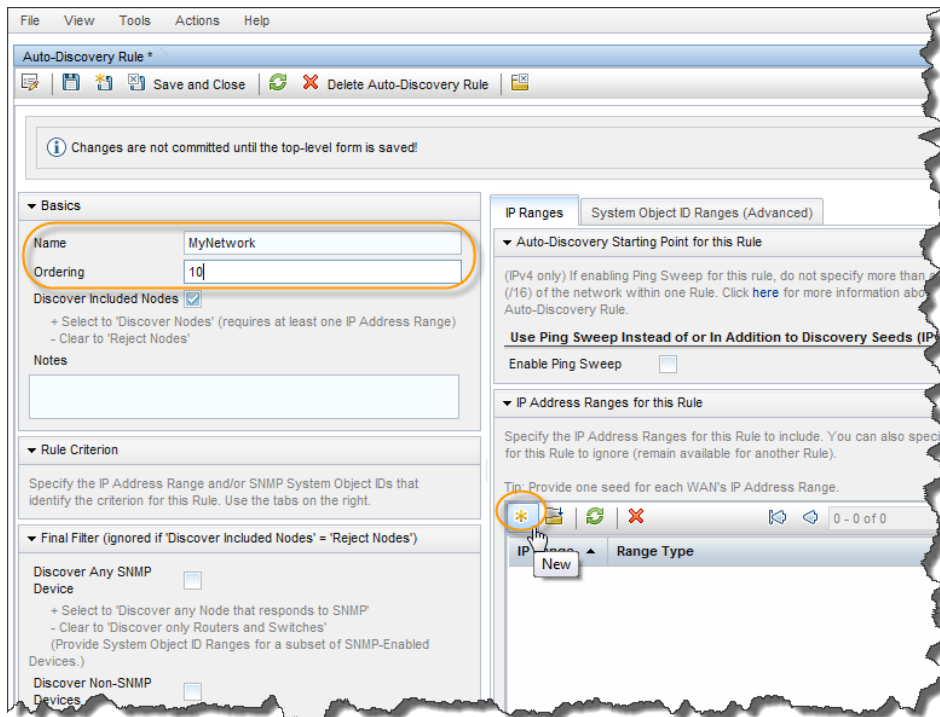
Figure 11: Discovery Configuration: Auto-Discovery Rules


3. Fill out the Basics section

Tip: NNMi uses the **Ordering** attribute value to prioritize multiple Auto-Discovery Rules. This example uses only one Auto-Discovery Rule.

Figure 12: Auto-Discovery Rule: Ordering Attribute

4. Click the ✳ icon to open an entry screen for the IP Range in this rule.

5. In the **IP Range** text box, enter the IP range you want to discover. Notice that you can enter both inclusive rules (`Include in rule`) and exclusive rules (`Ignored by rule`). The exclusive rules take priority over the inclusive rules.

Figure 13: Auto-Discovery IP Range
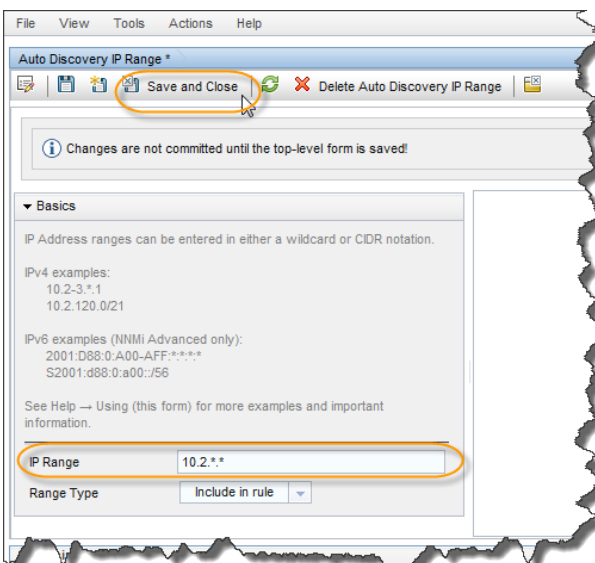
6. Click ⊞ **Save and Close** on this form as well as on the Auto-Discovery Rule form to save your changes.

This example does not use the ping-sweep feature.

Tip: If you choose to use the ping-sweep feature in your environment, NNMi sweeps across a maximum of a class B network (for example, 10.2.*.*) for each Auto-Discovery Rule.

Note the following:

- By default, NNMi discovers only routers and switches within the defined IP address range. To discover nodes beyond switches and routers, add system object ID ranges that include your other devices.

- If a node has multiple addresses, such as a router, then only one of the addresses must fall within the IP range. This address does not need to be the loopback address. NNMi might discover more nodes than you initially expect if you enter addresses other than the loopback addresses.

You now have one Auto-Discovery Rule defined. In most cases you only need one Auto-Discovery Rule since each rule can be quite complex.

Next, this example explains how to add a seed node.

Tip: It is better to add a router as a seed rather than a switch because routers provide a larger set of addresses for NNMi discovery.

1. From the workspace navigation panel, select the **Configuration** workspace, expand the **Discovery** folder, and then click **Seeds**.
2. Click the ✳ icon to create a new seed.
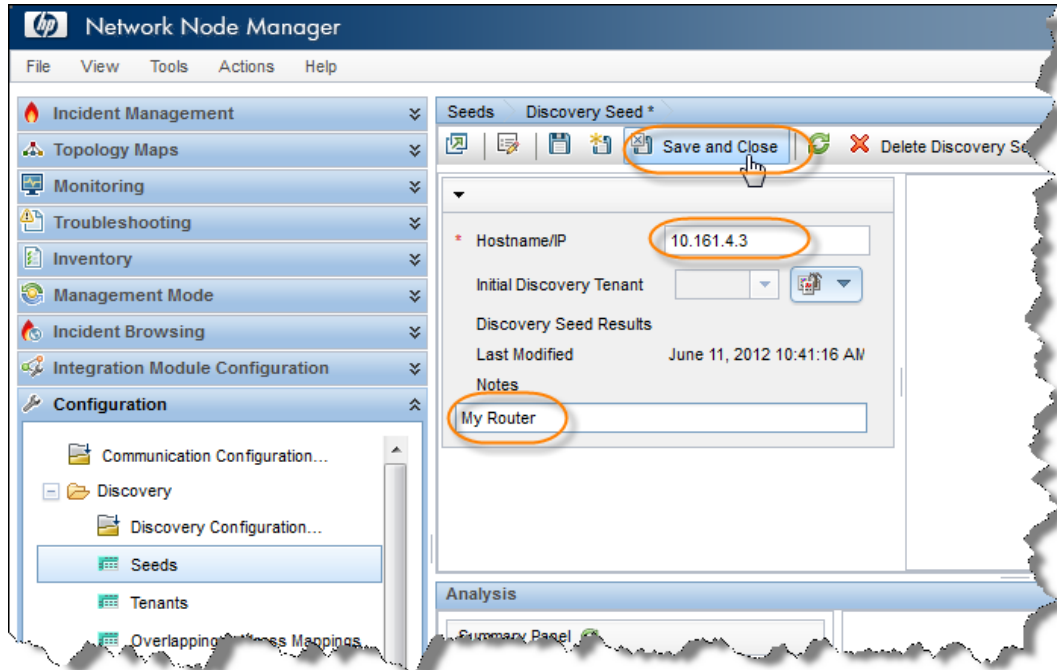
Figure 14: Discovery: Seeds
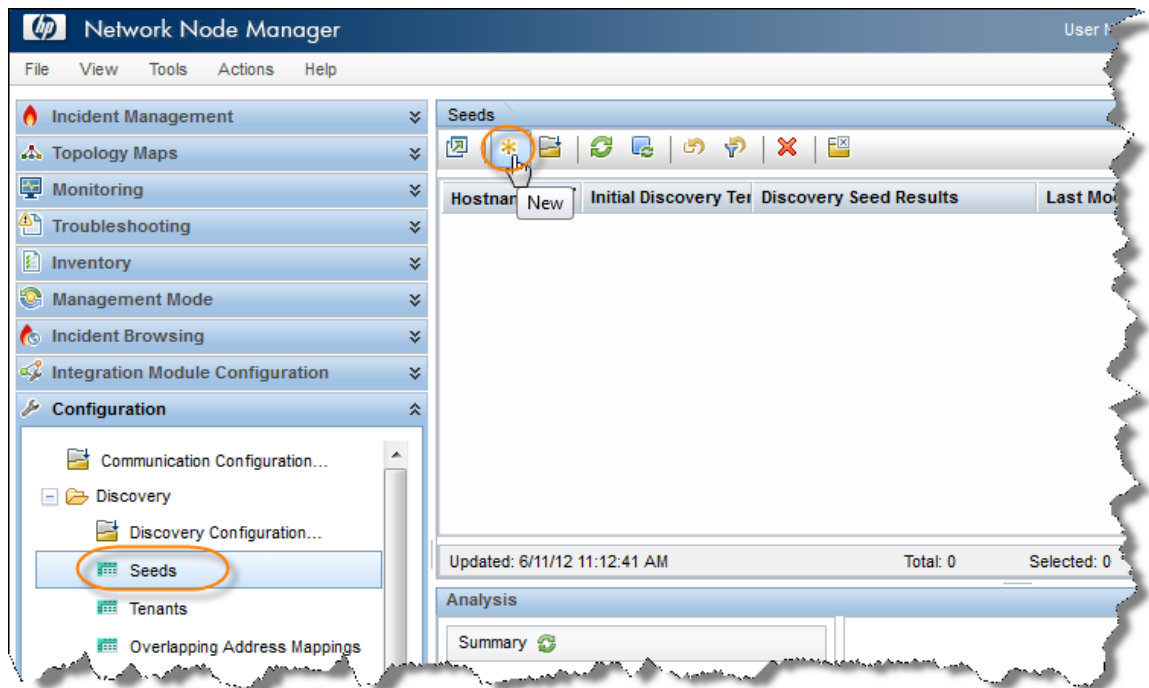
3. In the **Discovery Seed** form, enter the hostname or IP address and any Notes, as desired, and then click 📄 **Save and Close**.

Figure 15: Seeds: Discovery Seed

Tip: Examine the Discovery Seed Results column in the Seeds table to determine the discovery status of each seed. As NNMi begins discovering the node, NNMi displays the progress as `In progress`. When the discovery completes, the Discovery Seed Results entry changes to `Node Created`.

Figure 16: Seeds: Discovery Seed Results
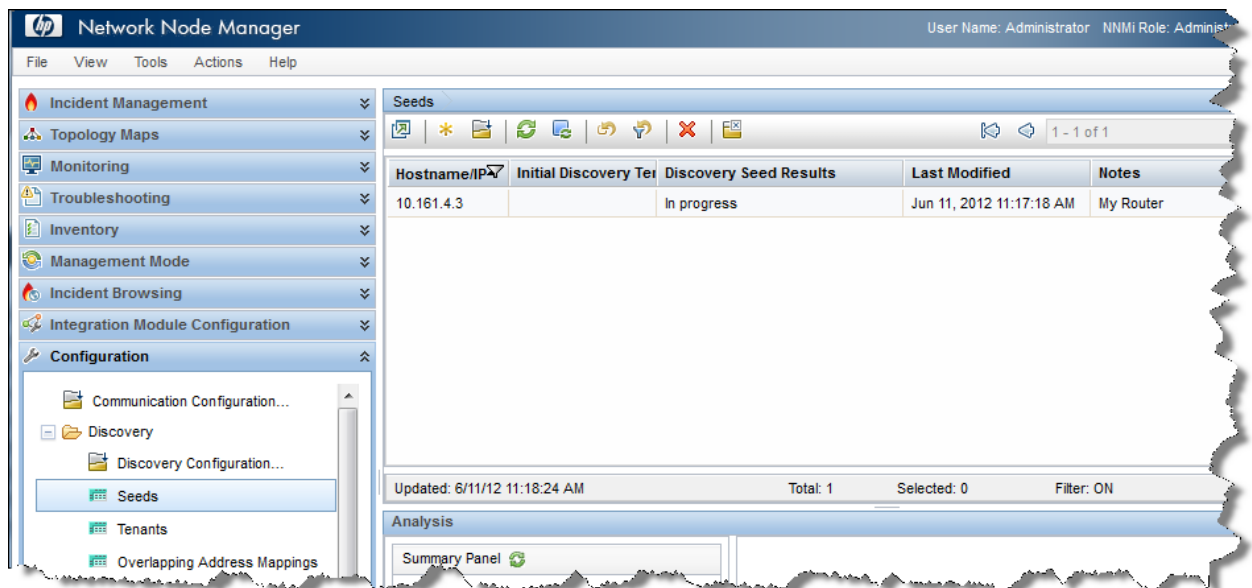
Tip: You can also load a list of seeds from a file using the `nnmloadseeds.ovpl` script. This script enables you to load a large number of seed nodes. If you use list-based discovery rather than Auto-Discovery Rules, you can load all of your nodes using the `nnmloadseeds.ovpl` script. See the `nnmloadseeds.ovpl` reference page or the UNIX manpage for more information.

When you use the Auto-Discovery method, Auto Discovery begins finding other switches and routers that have addresses within the address range specified in your Auto-Discovery Rule. Initially NNMi shows nodes without displaying status. Eventually NNMi shows a status for each discovered node.

The **Network Overview** map is useful to display discovery progress in smaller environments because the **Network Overview** map displays a limited number of nodes and connections.

Tip: Click **Refresh** on the **Network Overview** map to display the initial nodes.

Figure 17: Topology Maps: Network Overview
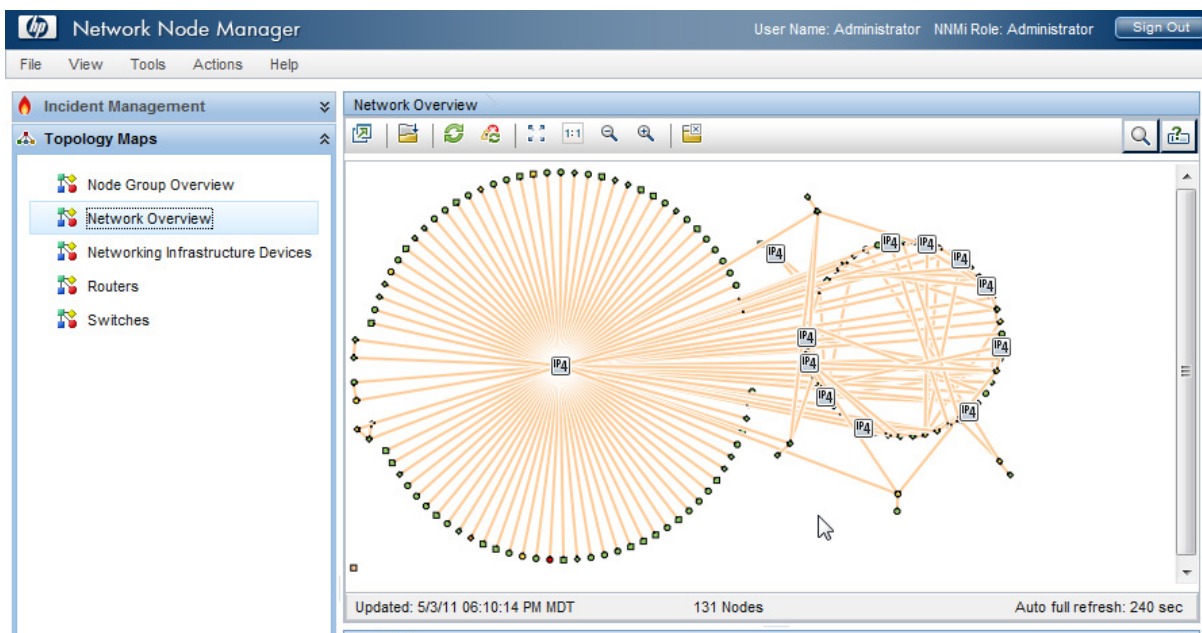
Configure Monitoring

Monitoring in NNMi is flexible and easy to configure. By default, NNMi uses SNMP polling rather than ICMP (ping) polling. The exception to this is non-SNMP nodes—NNMi polls these nodes using ICMP. You can enable ICMP polling more broadly if desired.

By default, NNMi polls *connected* interfaces. A connected interface in NNMi is an interface that is connected in the NNMi topology, which does not always include mapping to interfaces that have a wire connected.

Consider the following scenario:

- An access switch with 48 ports is connected to desktop computers and one uplink port.
- NNMi discovered the uplink node, but has not discovered any of the desktop computers.

In this case, only the uplink port will be considered *connected* to NNMi because it does not have a representation of the connection to the desktop computers. In most cases, this is the desired behavior. Usually, you will not want NNMi to notify you every time a computer is turned off for the evening.

In the following example, the `c2900xl-1` switch is an access switch with one uplink (`Fa0/2`). As shown in Figure 19: Node Form: List of Interfaces, only one interface is monitored.
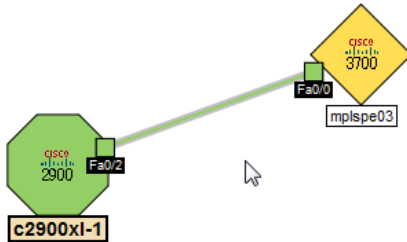
Figure 18: Map View: One Interface Monitored

Figure 19: Node Form: List of Interfaces

The second default behavior applies to routers. For routers, NNMi monitors most interfaces that host IP addresses. NNMi assumes that if an administrator takes the time to configure an IP address on an interface, it is desirable to monitor that interface. In some cases, NNMi models these interfaces as being connected; however, in other cases, NNMi models these interfaces as being unconnected. An example of this is a router that has an interface that connects to a WAN cloud. NNMi might not discover and model the connection to the cloud, but NNMi monitors the router interface by default.

When modifying this default behavior, note the following:

- NNMi enables you to modify monitoring settings in high volume.

- NNMi does this by using filters to apply monitoring to individual nodes, interfaces, and addresses. These filters are the same filters available for the user interface.

- Although this document focuses on nodes and interfaces, NNMi monitors additional entities such as Fans, and HSRP groups.

Consider the following scenario:

- Interfaces on a subset of nodes have an `IfAlias` that begins with `tunnel to`.

- You determine that NNMi needs to monitor these interfaces if their speed is 9 Kbs.

Using NNMi you can create a filter to identify any interfaces that match these criteria. After creating this filter, you apply monitoring settings to these interfaces.
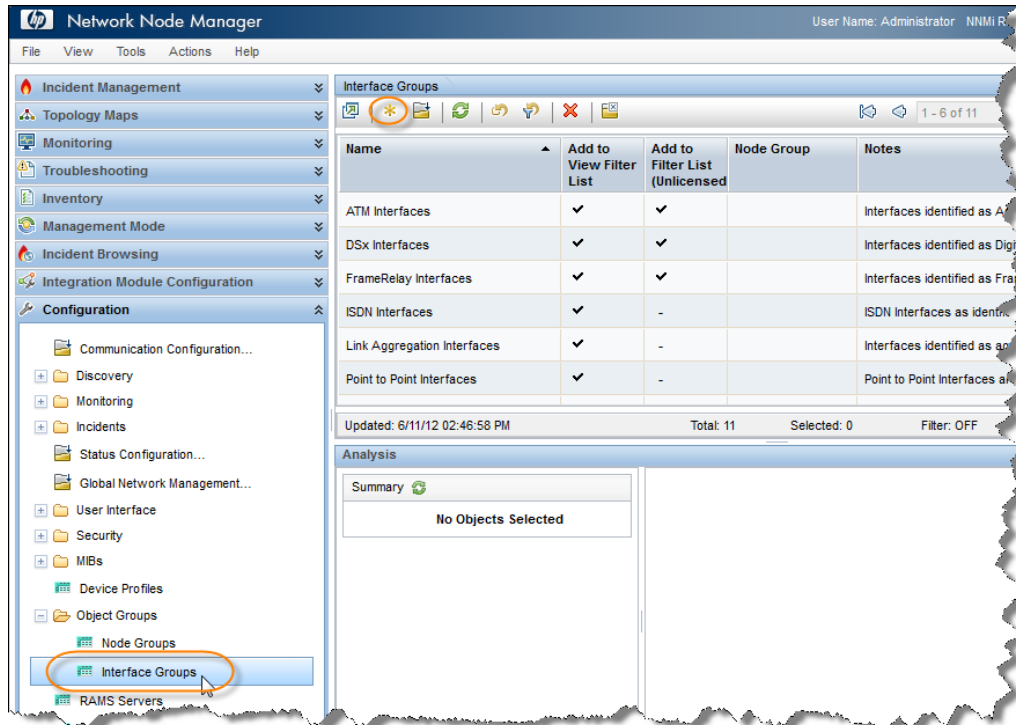
Figure 20: Node Form: Apply Monitoring Settings
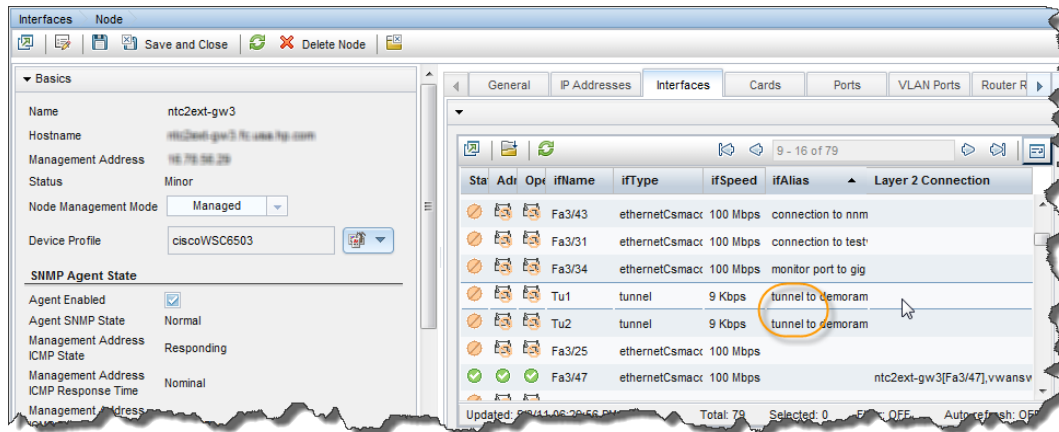
Create an Interface Group for Monitoring

NNMi enables you to create groups of nodes and interfaces. To create an Interface Group, follow these steps:

1. From the workspace navigation panel, select the **Configuration** workspace, and then click **Interface Groups**.

Figure 21: Configuration: Interface Groups

2. Click the ✳ icon to create a new Interface Group.

3. Enter `Important 9kbs Tunnels`, or some other descriptive name, in the **Name** text box.

Tip: Do not restrict this Interface Group to a specific Node Group; although often, you will do so.

4. Click the **Additional Filters** tab to access the **Filter Editor** used to define the filter logic.

You define a filter expression by selecting an `Attribute`, an `Operator` and a `value`. You can use the `like` operator along with an asterisk for variable matching.

In this example, use an AND condition for the two attributes.

Tip: If you encounter problems when defining your logic, close the form without saving it to return to the last saved value. Then re-open the form and begin again.

Note: If you define an IfType filter (on the **IfType Filters** tab), then it is always logically AND'ed with the filters on the **Additional Filters** tab.

Figure 22: Interface Groups: Save
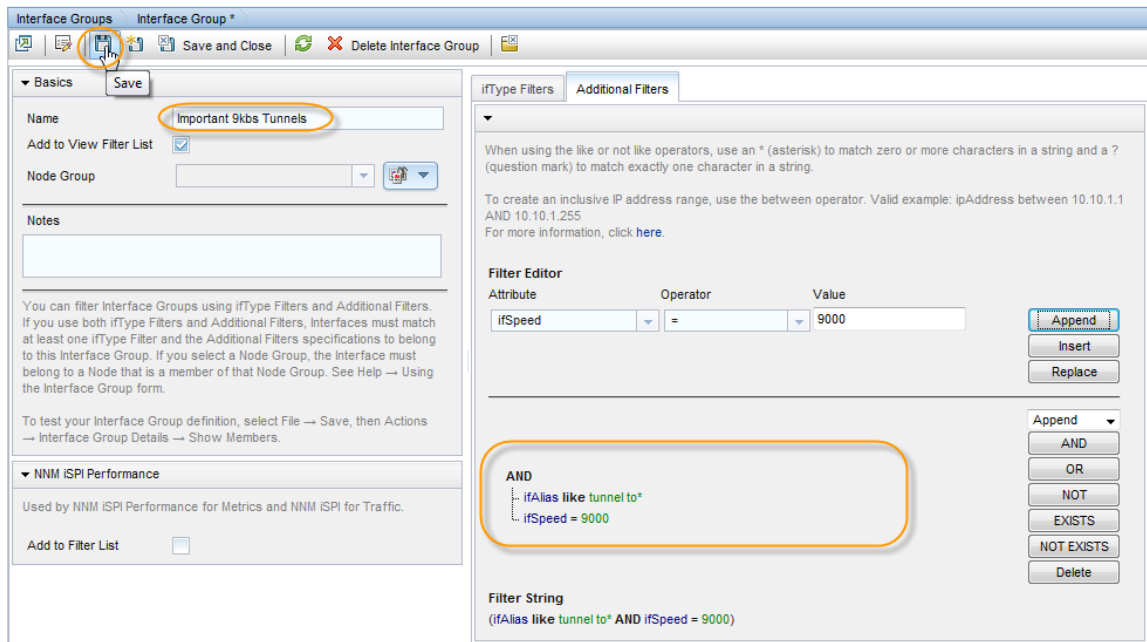
5. After you specify your filter, save the filter, but do not close it.

6. Verify that the filter works as expected using the **Actions > Show Members** menu item.

   NNMi displays all items that pass the filter criteria.

Figure 23: Actions: Show Interface Group Members
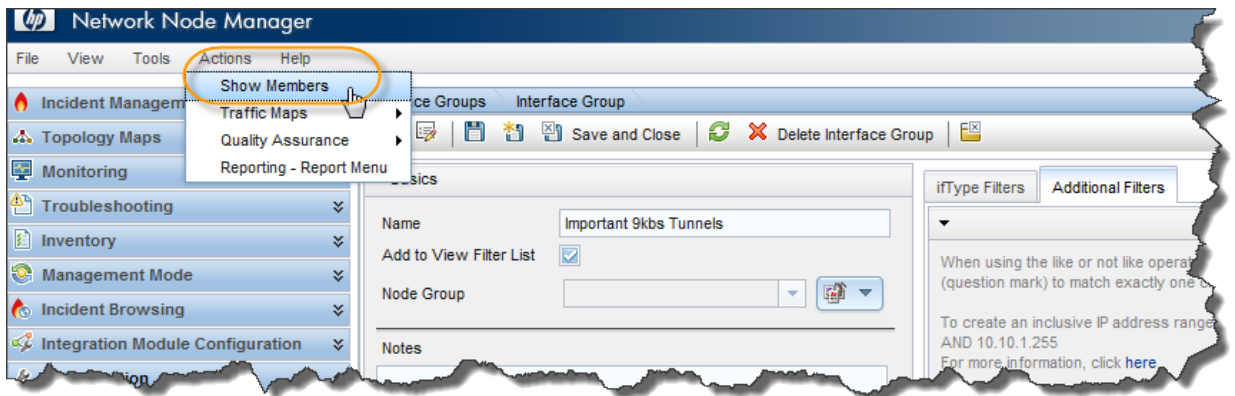
7. Verify the results. In this example, you can see that the filter matched a number of interfaces in the network. NNMi is already monitoring some of them.

Figure 24: Interfaces: Interface Group Filter Results

Apply Monitoring to an Interface Group

To monitor the interfaces defined by the filter just created, apply monitoring to this Interface Group. You can apply monitoring to both Node Groups and Interface Groups.

Note: NNMi considers an interface setting to be a higher priority than a node setting.

1. From the workspace navigation panel, select the **Configuration** workspace, and then click **Monitoring Configuration**.

Figure 25: Monitoring Configuration
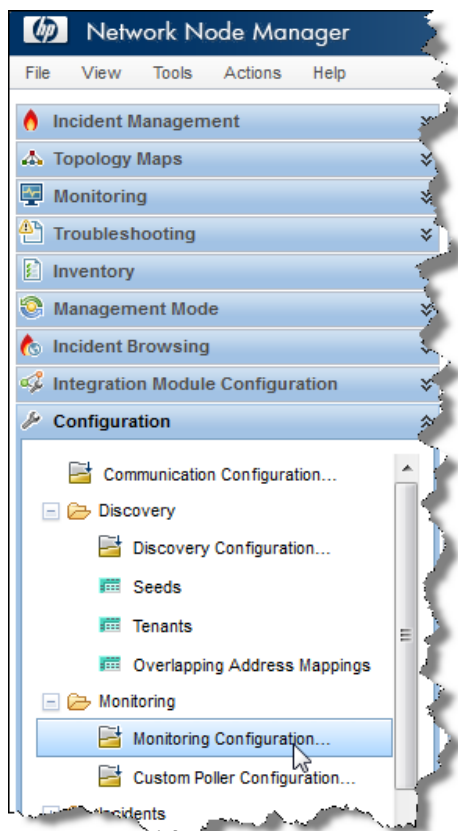
2. Click the **Interface Settings** tab.

Tip: Take note of the current Ordering values. These define priority if an interface belongs to multiple groups.

In this example, the highest priority is `100`.

Figure 26: Monitoring Configuration: Interface Settings Tab
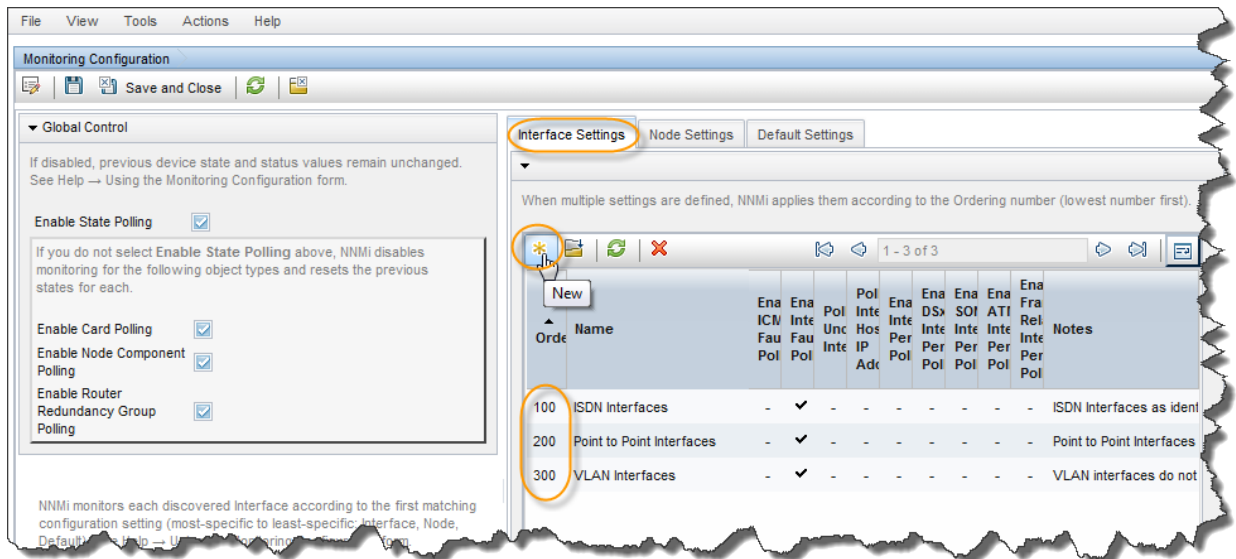
3. Click the ✳ icon.

4. Enter an **Ordering** value that configures this setting to have a higher priority than other settings. This ensures that these interfaces get polled. NNMi considers lower numbers to be higher priority. You also want to choose an **Ordering** value that takes into consideration future configurations. For example, if you set this number to `1`, that sets the highest priority possible and limits your future entries. For this example, enter `50`.

5. Extend the monitoring scope. To monitor these interfaces regardless of whether they are connected, click all the check boxes in the Extend the Scope of Polling Beyond Connected Interfaces area of the form.

6. Use the **Quick Find** feature to select your newly created Interface Group. Then click ⊠ **Save and Close**.

7. Click ⊠ **Save and Close** at the top level **Monitoring Configuration** form to save your changes.

Figure 27: Interface Settings: Save and Close
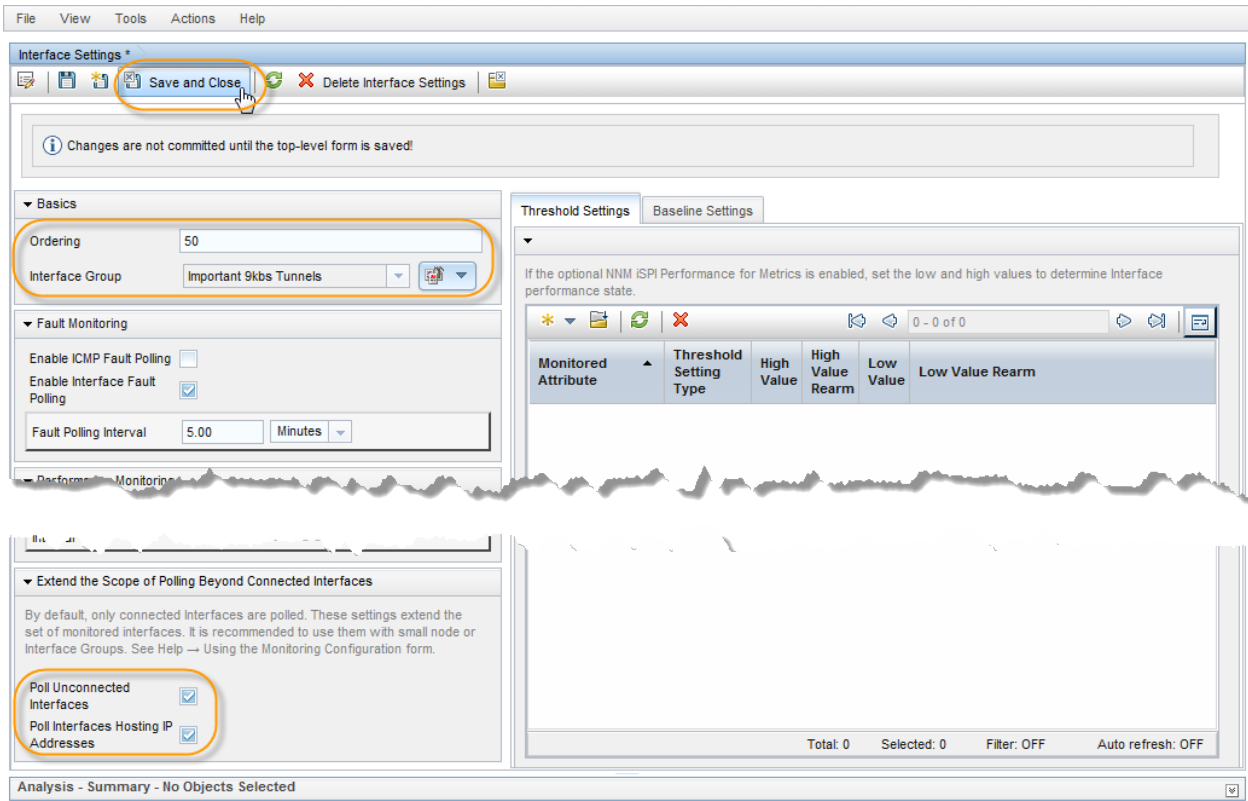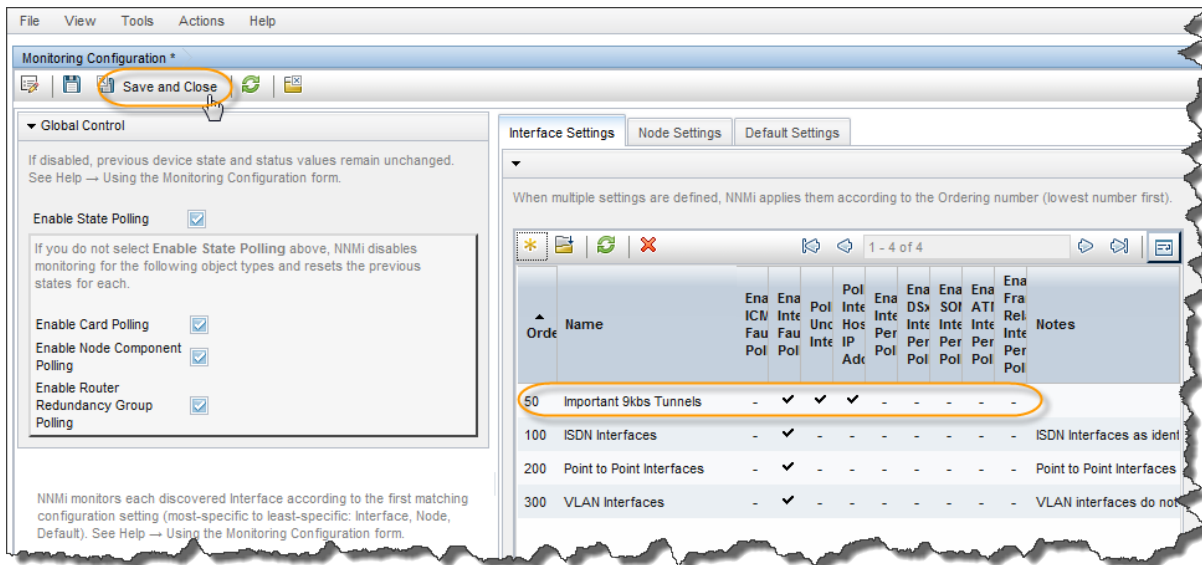

Figure 28: Monitoring Configuration: Save and Close

Now that you have a monitoring setting that applies to everything in this Interface Group, NNMi uses SNMP to monitor any interface that matches the `Important 9kbs Tunnels` filter.

Test the Monitoring Settings

You can test your new monitoring settings in many different ways. For this example, use the following steps:

1. From the workspace navigation panel, select the **Inventory** workspace, and then click **Interfaces**.

2. Use the drop-down menu to select the new Interface Group, `Important 9kbs Tunnels`.

This filters the table to only show the interfaces in this Interface Group.

Tip: You might notice that some of the interfaces have an Administrative State of Not Polled. It can take a few minutes for your Monitoring configuration changes to take effect. To manually force the interfaces to be polled, perform a Status Poll command on one of the nodes hosting these interfaces. You should see them all begin to acquire status.

To perform a Status Poll on a node:

1. From the workspace navigation panel, select the **Inventory** workspace, and then click **Nodes**.

2. Select the node you want to poll, and then use the **Actions > Polling > Status Poll** command to start the Status Poll.

Figure 29: Interfaces: Important 9kbs Tunnels Filter

Open one of the interfaces highlighted in the previous figure and check the monitoring settings to confirm that your monitoring settings are working properly.

To check monitoring settings for an interface:

1. Double-click the interface.

2. Click **Actions > Configuration Details > Monitoring Settings** to view the monitoring configuration for the selected interface.

Figure 30: Actions: Monitoring Settings
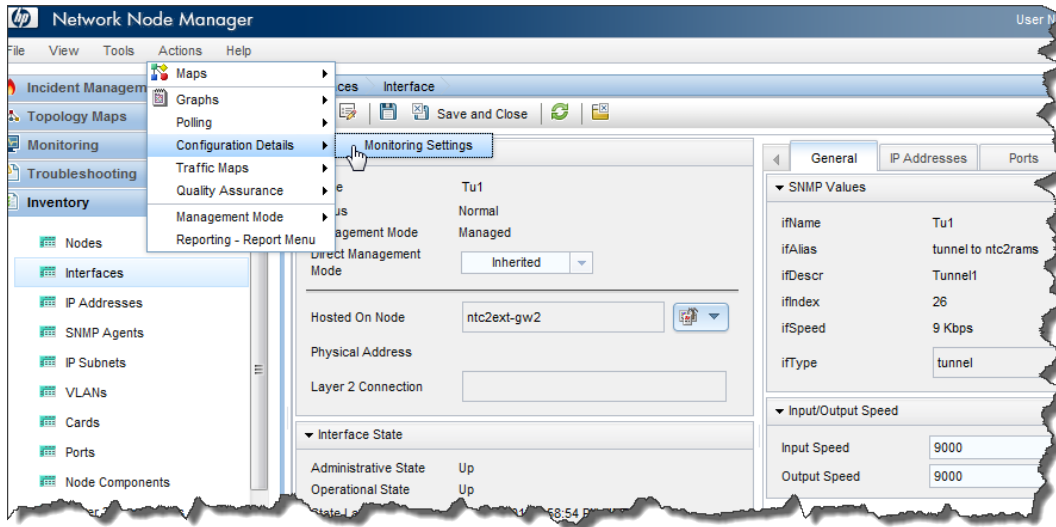
This example report confirms that the monitoring settings are working properly:

First, you can see that NNMi applied the monitoring settings for the `Important 9kbs Tunnels` group to this interface. This shows you that the monitoring settings are properly associated with this interface.

Second, you can see that NNMi has `Fault SNMP Polling Enabled` set to true. This indicates that the new monitoring settings are successfully applied to the `Important 9kbs Tunnels` Interface Group.

**Figure 31: Monitoring Settings Report: Interface**

Monitoring Exceptions

You can manually force an interface or node to be unmonitored.

From the **Interface** form, click **Actions > Management Mode > Not Managed** to switch to unmanaging the interface.

NNMi no longer monitors this interface regardless of the monitoring settings.

Figure 32: Actions: Management Mode: Not Managed
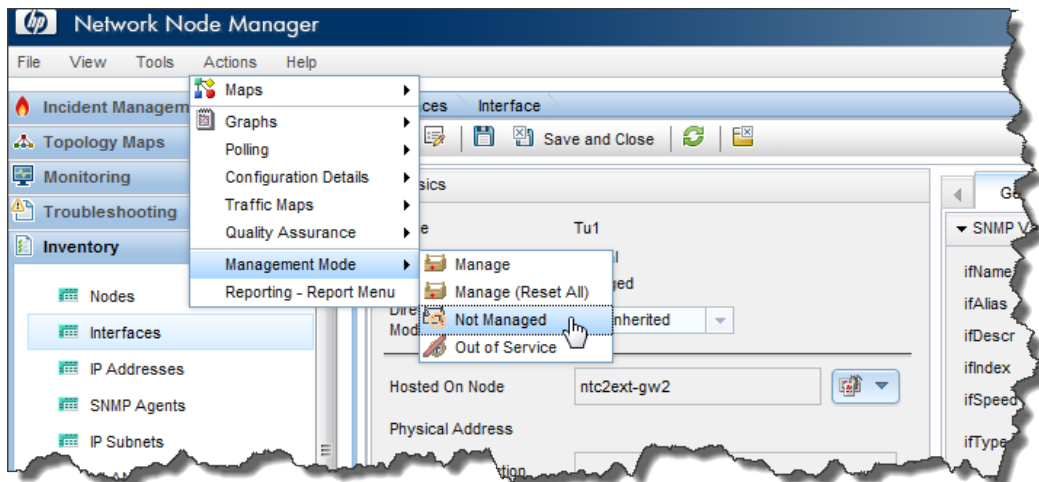
NNMi does not presently have the same approach that NNM used to force an interface to be unmonitored. Currently, unmanaging an interface is only a *negative override.*

See *Forcing an Interface to be Polled,* available at http://h20230.www2.hp.com/selfsolve/manuals, to force NNMi to monitor an interface.

Configure Incidents, Traps, and Automatic Actions

Configure Incidents

With NNMi, you can change certain aspects of an incident. Some examples include enabling an incident, formatting a message, enabling de-duplication, and enabling rate correlation.

This example describes how to enhance the `InterfaceDown` (Interface Down) incident to include the `Interface Alias` in the message.

1. From the workspace navigation panel, select the **Configuration** workspace, and then click **Incidents > Management Event Configurations**.

2. Double-click the `InterfaceDown` incident configuration.

Figure 33: Configuration: Management Event Configurations

3. Before continuing, see "*Valid Parameters for Configuring Incident Messages*" in the NNMi help to view the possible arguments that can be added to a message format. In this example, add the argument `$ifAlias` to the incident message as shown in the following example.

Figure 34: Management Event Configuration: Message Format

4. Change the **Author** to **Customer** using [icon] ⊤ **Quick Find**.

5. Finally, click [icon] **Save and Close** on this form and in the **Management Event Configuration** form.

As shown in the following **Open Key Incidents** view example, all `InterfaceDown` incidents show the `$ifAlias` parameter.

Note: If there is no alias on the interface, NNMi displays `null` for the alias.

Figure 35: Open Key Incidents

Configure Traps

Tip: See *Step- by-Step Guide to Incident Management,* available at http://h20230.www2.hp.com/selfsolve/manuals, for more details about working with traps in NNMi

Note: To receive a trap into the NNMi Incident Browser, you must load the MIB that contains the trap definitions into NNMi.

For this example, you need to load three MIBs to satisfy the dependencies. You first load the ruggedcom.mib file, followed by the rcsysinfo.mib file. Then you can load the traps from the ruggedcomtraps.mib file. Use the **nnmloadmib.ovpl** command to load the MIBs into NNMi.

Note: You can also use the NNMi console to load MIBs.

To load MIBs using the command line:

1. Run the **nnmloadmib.ovpl -load ./ruggedcom.mib** command. This loads the ruggedcom.mib definitions.

2. Run the **nnmloadmib.ovpl -load ./rcsysinfo.mib** command. This loads the rcsysinfo.mib definitions.

3. Run the **nnmloadmib.ovpl -load ./ruggedcomtraps.mib** command. This loads the ruggedcomtraps.mib file.

Next, verify that the MIBs are loaded:

1. From the workspace navigation panel, select the **Configuration** workspace, and then click **MIBs > Loaded MIBs**.

Notice the newly loaded Rugged Com MIBs.

2. Take note of the traps module (RUGGEDCOM-TRAPS-MIB). You will need this for the next command.

Figure 36: Configuration: Loaded MIBs
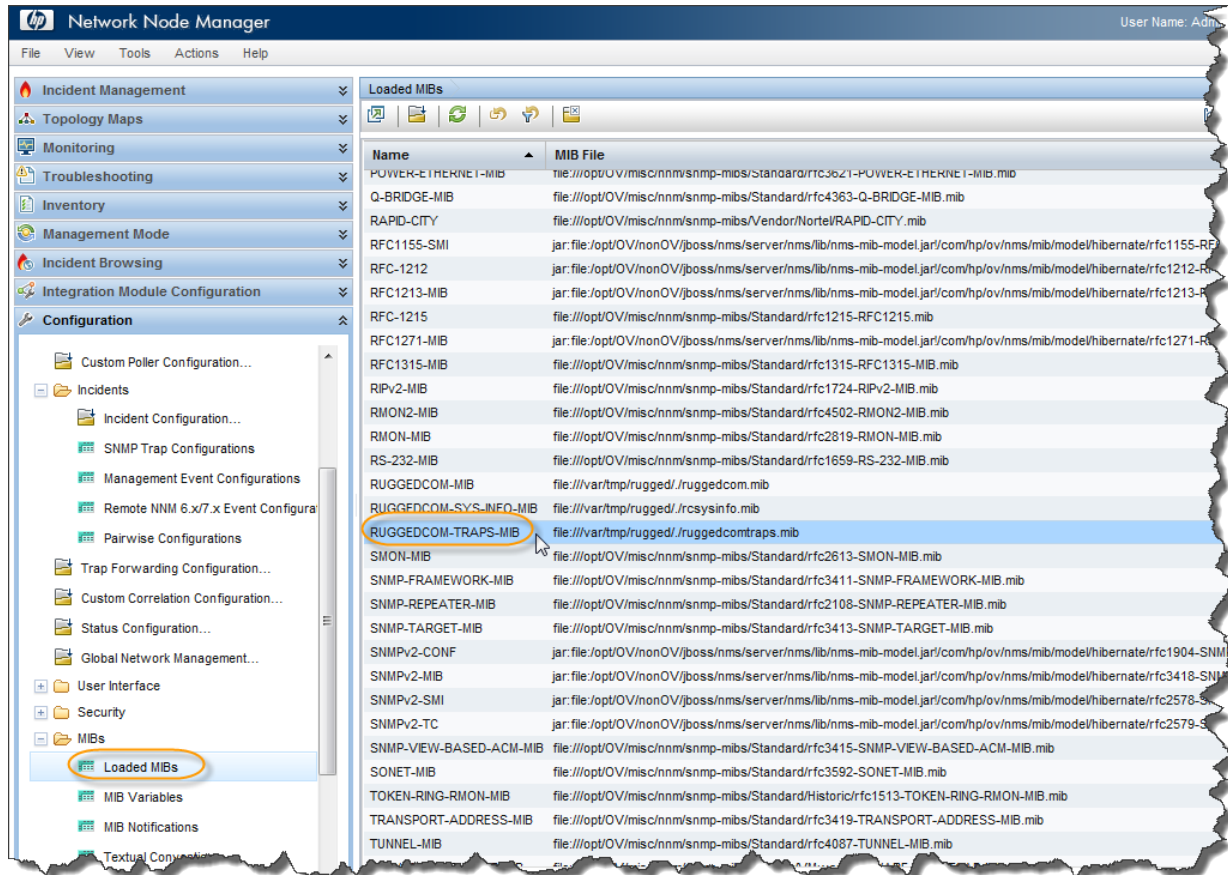
3. Run the `nnmincidentcfg.ovpl –loadTraps RUGGEDCOM-TRAPS-MIB` command to load the traps from this module. You should see output similar to the following:

```
SNMP trap(s) from mib module loaded: RUGGEDCOM-TRAPS-MIB.
Number of traps: 5.
The following traps were added to incident configuration:
cfgChangeNoRevTrap - .1.3.6.1.4.1.15004.5.5
cfgChangeTrap - .1.3.6.1.4.1.15004.5.4
powerSupplyTrap - .1.3.6.1.4.1.15004.5.2
swUpgradeTrap - .1.3.6.1.4.1.15004.5.3
genericTrap - .1.3.6.1.4.1.15004.5.1
```

You now have four new traps defined in NNMi. To view them:

1. From the workspace navigation panel, select the **Configuration** workspace, and then click **Incidents > SNMP Trap Configurations**.
2. Sort the traps by **SNMP Object ID**.

Notice that all of the traps are loaded as *enabled*. You may want to disable all but the ones you specifically want to receive. You may want to make configuration modifications at this time.

Figure 37: Configuration: SNMP Trap Configurations
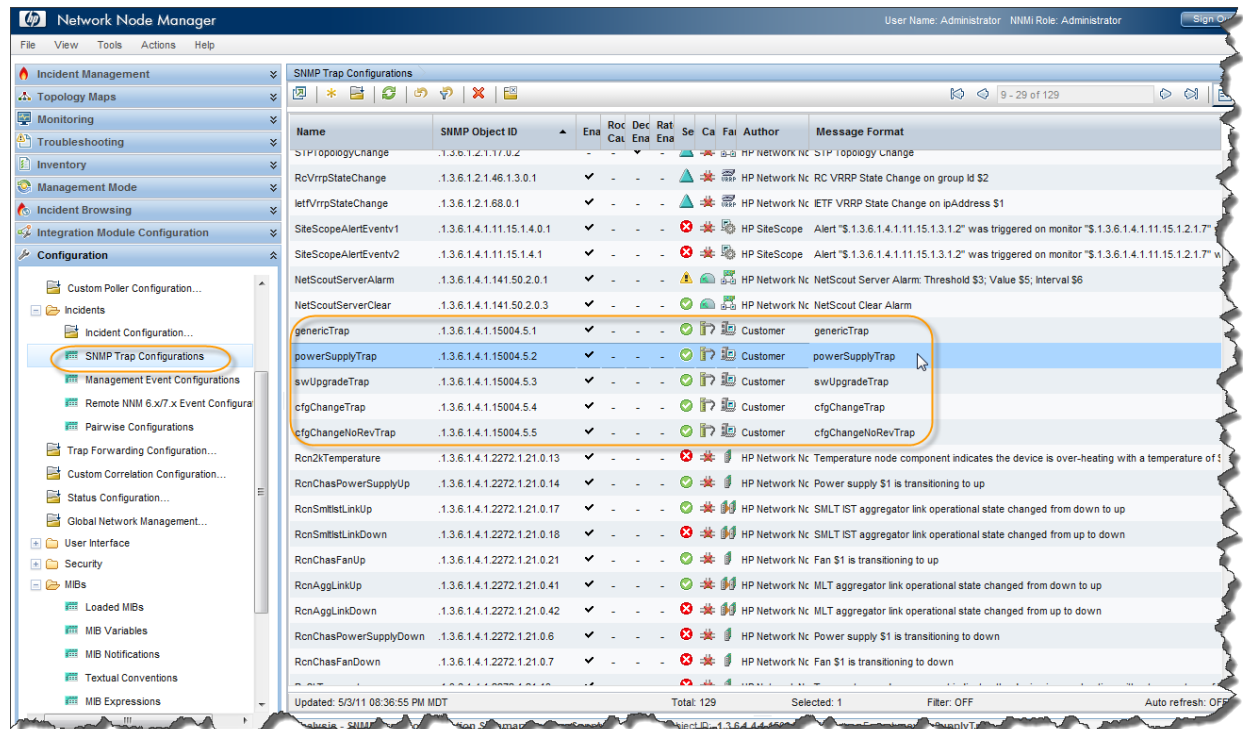
Configure Automatic Actions

You can configure automatic actions for incidents. Usually you do this for only management events rather than for SNMP traps, because it is hard to predict the rate and volume of traps. NNMi automatic actions can be executable commands, command line scripts, or Python scripts. The Python scripts execute within NNMi's Java virtual machine (JVM) so they execute quickly. Since NNMi uses a Java interpreter for Python, NNMi refers to these scripts as `Jython`.

In NNMi, actions are based on Lifecycle Sate changes for incidents. You can configure NNMi to take one action when an interface goes down and another action when the interface comes back up again. To do this, configure both actions on the `InterfaceDown` incident, but associate one action with the Lifecycle State set to `Registered` and the other action with the Lifecycle State set to `Closed`. Usually NNMi does not generate an associated `up` incident.

Note: When NNMi generates an incident, it assigns the `Registered` state to the incident.

To configure NNMi to run a Perl script when it receives a Node Down incident, do the following:

1. Place your script in the `actions` directory.

   Note: For security reasons, you must be `root` or `administrator` to access this directory.

   For this example, assume the `actions` directory appears in the following location:

   - *Windows*: `\Documents and Settings\All Users\Application Data\HP\HP BTO Software\shared\nnm\actions`
   - *UNIX*: `/var/opt/OV/shared/nnm/actions`

The `actions` directory can be in a different location depending on how you installed NNMi. For this example, the script is named `writelog.ovpl`. Copy this script into the `actions` directory. Make sure that your script is executable.

2. To associate this script with an action on this incident:

   a. From the workspace navigation panel, select the **Configuration** workspace.

   b. Click **Incidents > Management Event Configuration**.

   c. Double-click the `NodeDown` incident.

Figure 38: Management Event Configurations: NodeDown Incident

3. Change the **Author** to **Customer**, click the **Actions** tab, and click the ✻ icon.

Figure 39: Management Event Configuration: Actions Tab

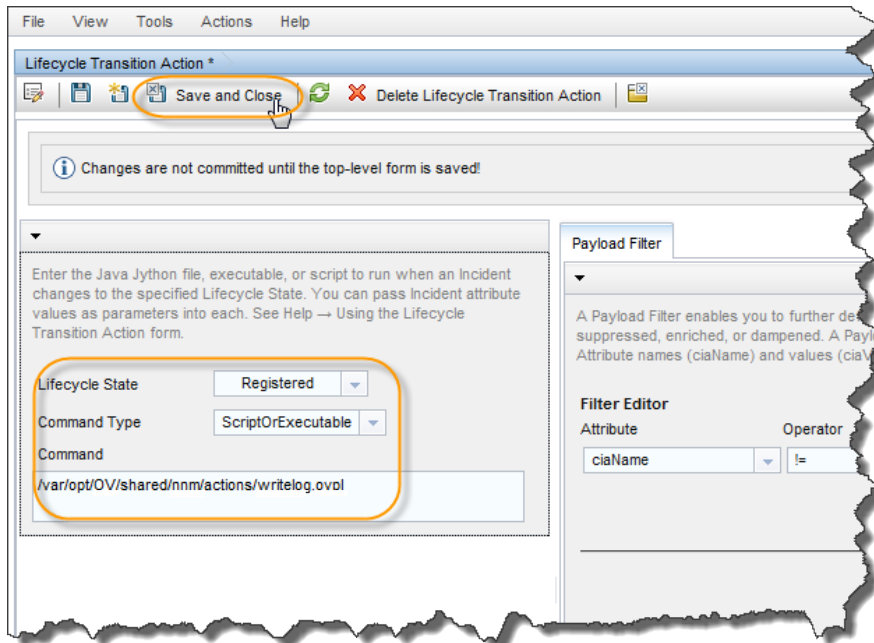4. Select the appropriate **Lifecycle State** (Registered in this example).

5. Set the **Command Type** to ScriptOrExecutable.

6. Enter the name of the command, including the complete path to the executable, and then click ⊠ **Save and Close**.

Figure 40: Lifecycle Transition Action

7. Click the **Enabled** check box to enable the action.

Figure 41: Management Event Configuration: Actions Tab: Enable Action

Next, you need to test the action. The easiest way to do this is to look for a previous occurrence of the
NodeDown incident:

1. From the workspace navigation panel, select the **Incident Browsing** workspace, and then click
    Closed Key Incidents.

Figure 42: Incident Browsing: Closed Key Incidents View
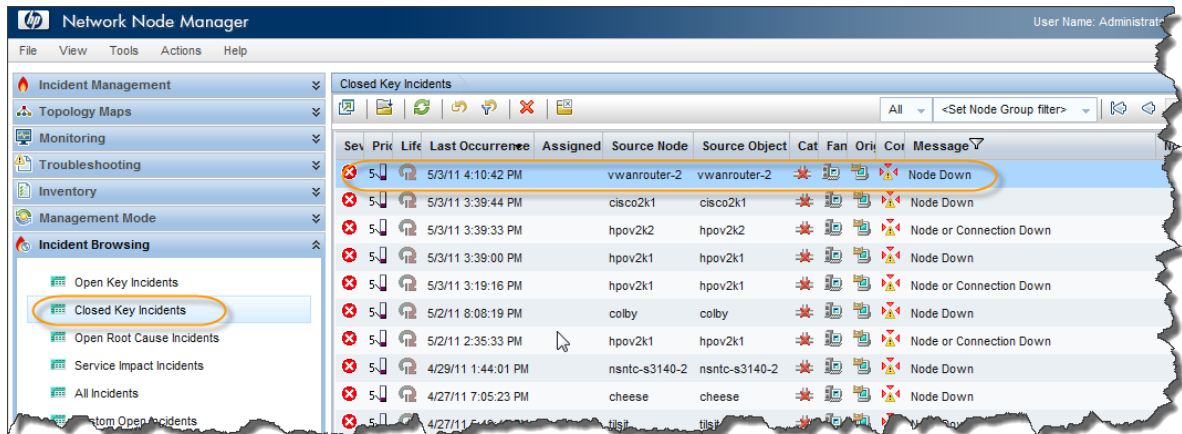
2. Open a NodeDown incident that NNMi closed.

In this example Closed means that the interface is back up. NNMi automatically closes an incident
when a fault is cleared. (You can re-open the incident by setting the **Lifecycle State** to Registered.
After you take this action, NNMi behaves as if the incident is opened for the first time when executing
actions.)

3. Set the **Lifecycle State** to Registered.

This causes your action to execute after you save this form (saving the **Lifecycle State** change). If you
change the **Lifecycle State** without saving the change, NNMi takes no action.

4. Click 💾 **Save** after each **Lifecycle State** change.

Figure 43: Incident Form: Registered Lifecycle State
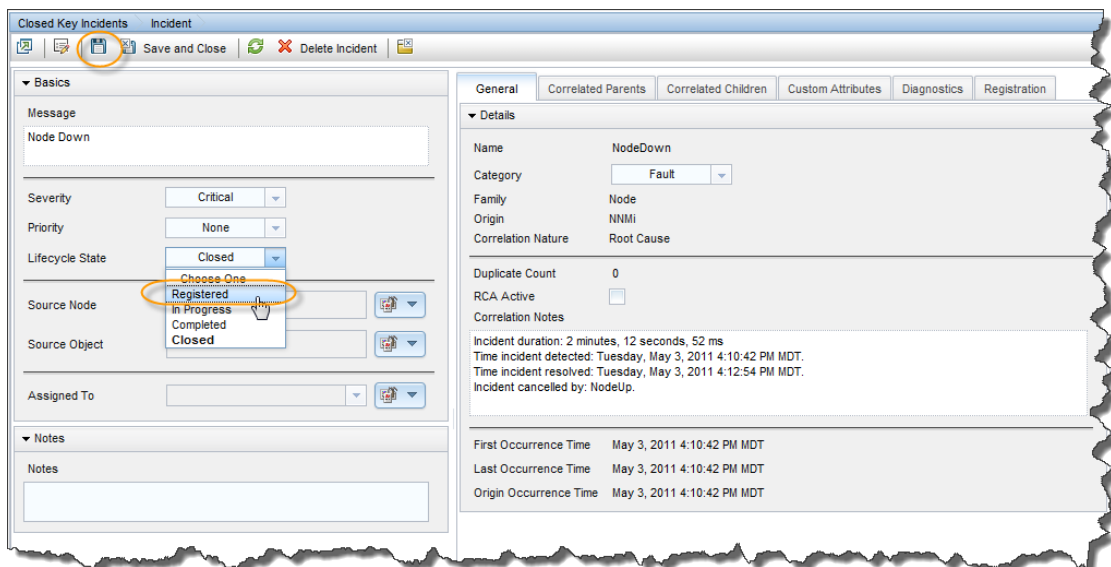
After saving your change, verify your action's results. In this case, look at the log file associated with this script. After you finish testing, set the **Lifecycle State** back to `Closed`, and then save the incident to return it to its original state.

Configure the NNMi Console

Configure Node Groups

To enhance diagnostics, create Node Group maps, which show the nodes contained in a Node Group.

See *A Practical Example of Using Node Groups* in the *HP Network Node Manager i Software Deployment Reference,* available at http://h20230.www2.hp.com/selfsolve/manuals, for more information about configuring Node Groups.

This example creates Node Groups for a few different subnets.

Tip: You want these Node Groups to refer to management addresses rather than addresses on the node. You also want these Node Groups to contain nodes based on names.

Note: The same node can be in multiple Node Groups.

The following diagram describes an example hierarchy of Node Groups:

Figure 44: Hierarchy of Groups
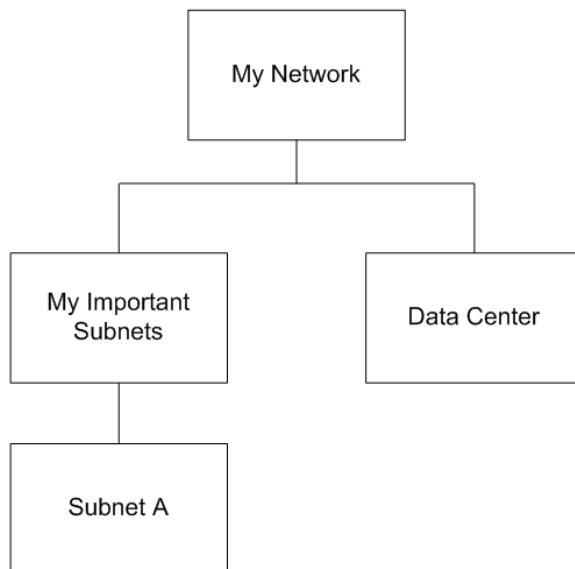
Subnet A = Management Address of 192.125.*.*

Data Center = Nodes that have a system name beginning with "data_center"

Note the following:
- Only the Subnet A Node Group and Data Center Node Group are populated with nodes. The My Important Subnets Node Group shows structure in the hierarchy and is populated only with a Child Node Group.
- It is easiest to work your way up the hierarchy.

1. Create the Subnet A Node Group as shown in the following example:

Tip: Notice the unique expression for IP address ranges.

Figure 45: Node Group: Basics
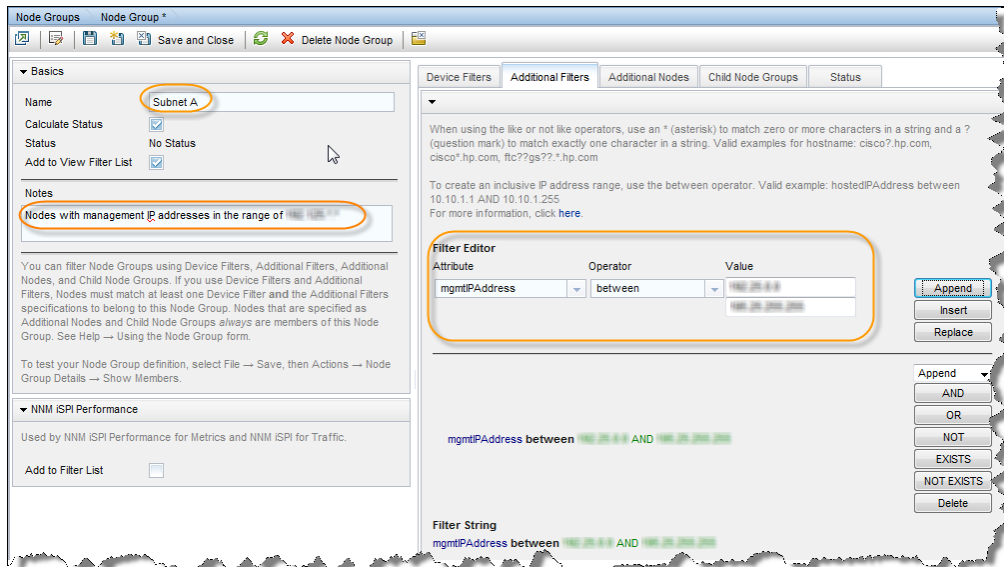
2. Next, create the Data Center Node Group.

Figure 46: Node Group: Additional Filters Tab
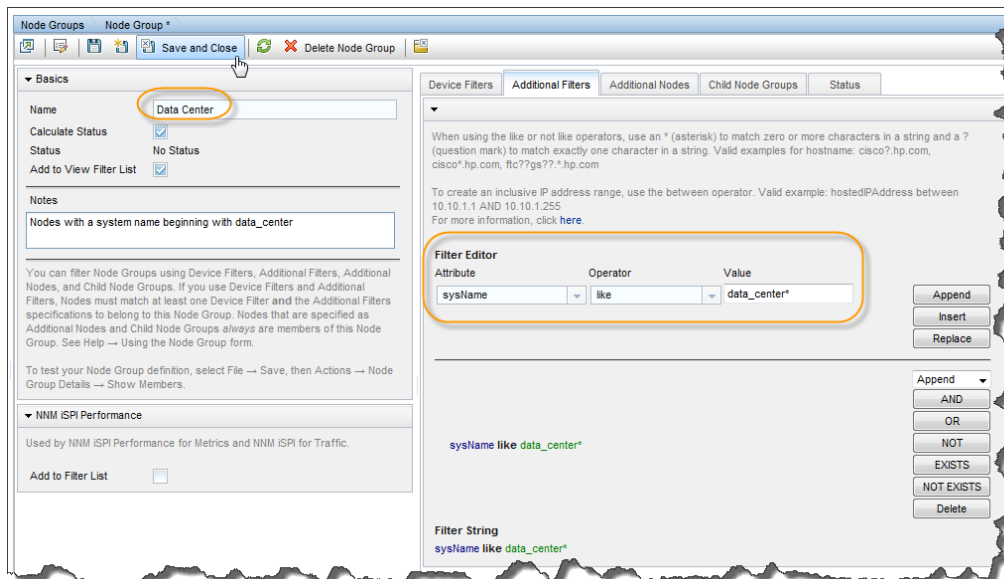
Next, create the Node Group called My Important Subnets:

1. On the Node Groups form, click the ✳ icon.
2. Enter `My Important Subnets` in the **Name** text box.
3. Click the **Child Node Groups** tab, and then click the ✳ icon.

Figure 47: Node Group: Child Node Group Tab
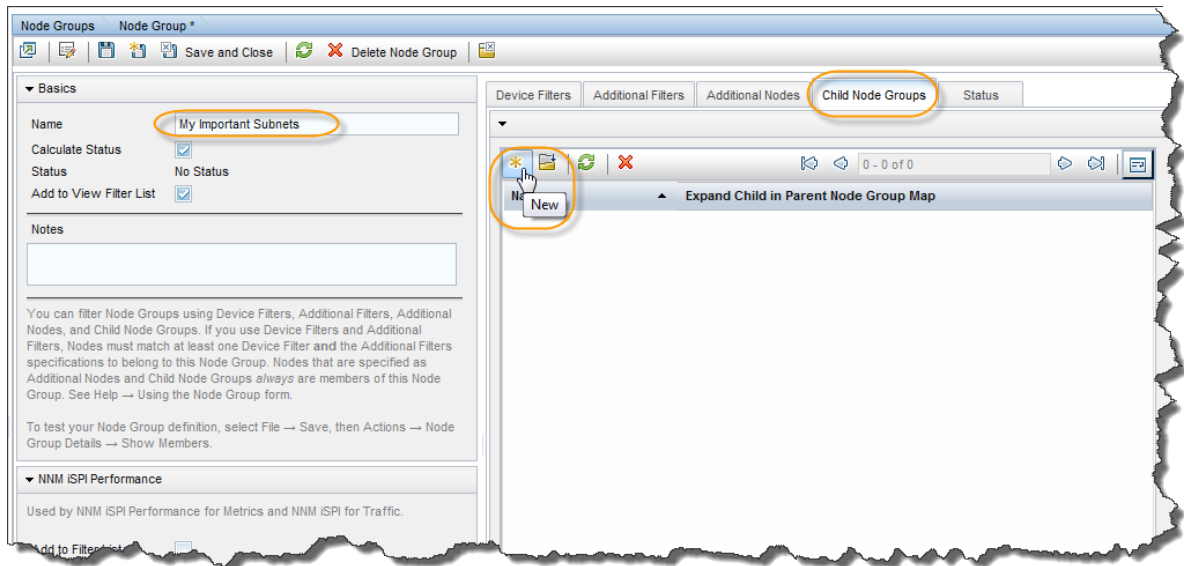
4. Click ![icon], and then click **Quick Find**. Click the **Subnet A** Child Node Group, and then click **OK**.

Figure 48: Node Group Hierarchy: Assign Child Node Group Name
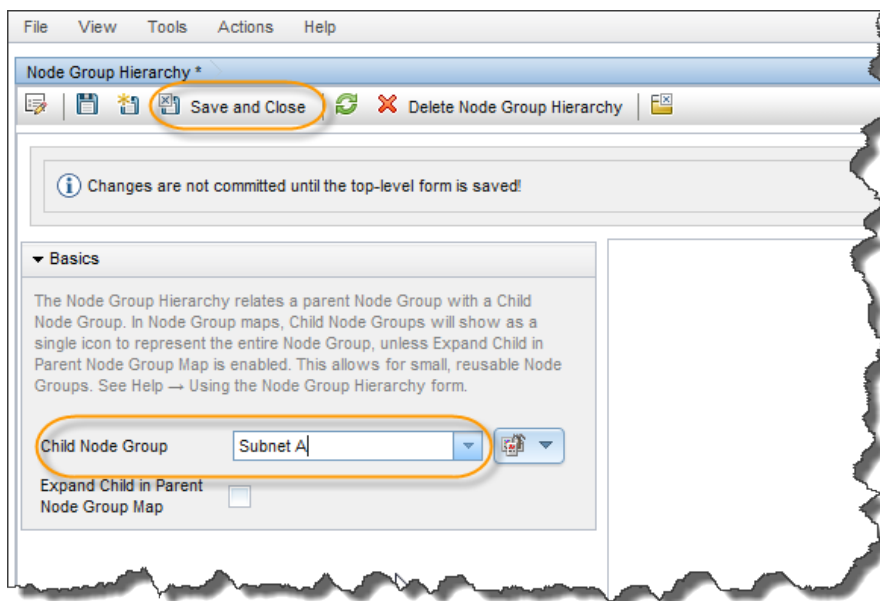
5. Click ⊠ **Save and Close**. You just created a Child Node Group, Subnet A, for the My Important Subnets Node Group.

Figure 49: Child Node Groups Tab: Save and Close
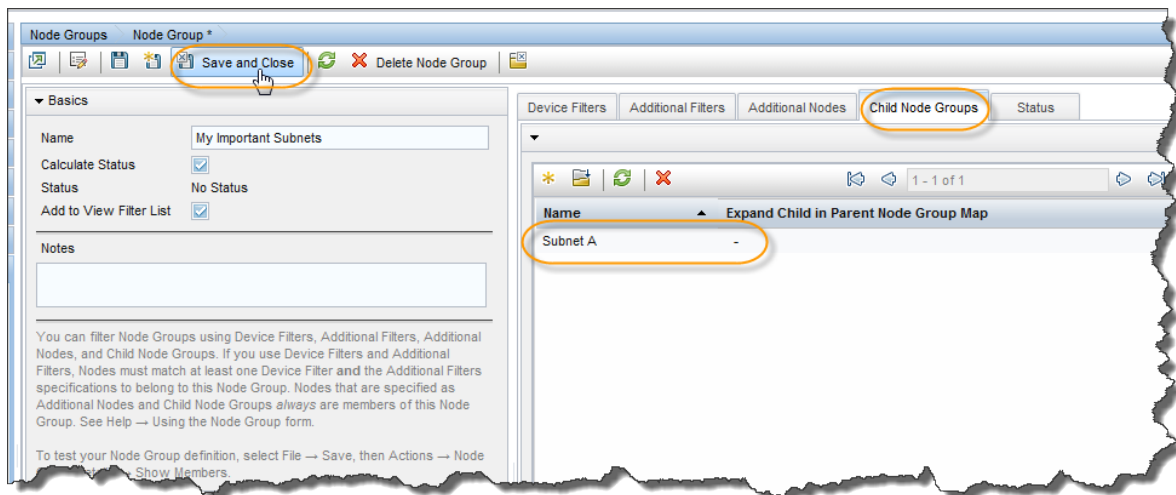
Finally, create the Node Group called My Network that includes the following Child Node Groups: Data Center and My Important Subnets.

Tip: Remember to test the membership after you save each Node Group by clicking **Actions > Node Group Details > Show Members**.

After you test the population of the Node Groups, create an initial instance of a map for each Node Group:

1. Click **Actions > Maps > Node Group Map** to open the map.

Figure 50: Actions: Map: Select Node Group Map
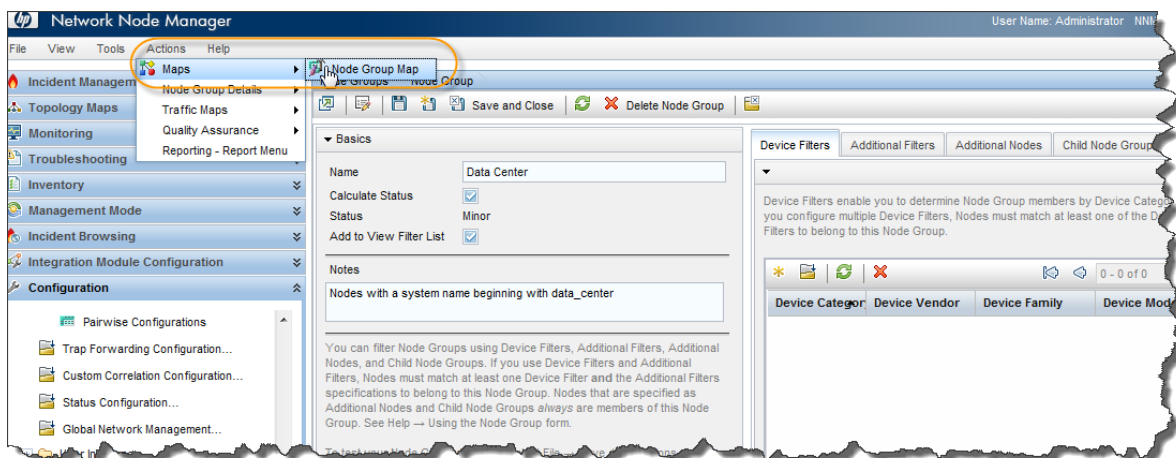
2. Click 🔲 **Save Layout**.

Figure 51: Node Group Map: Save Node Group Map Layout
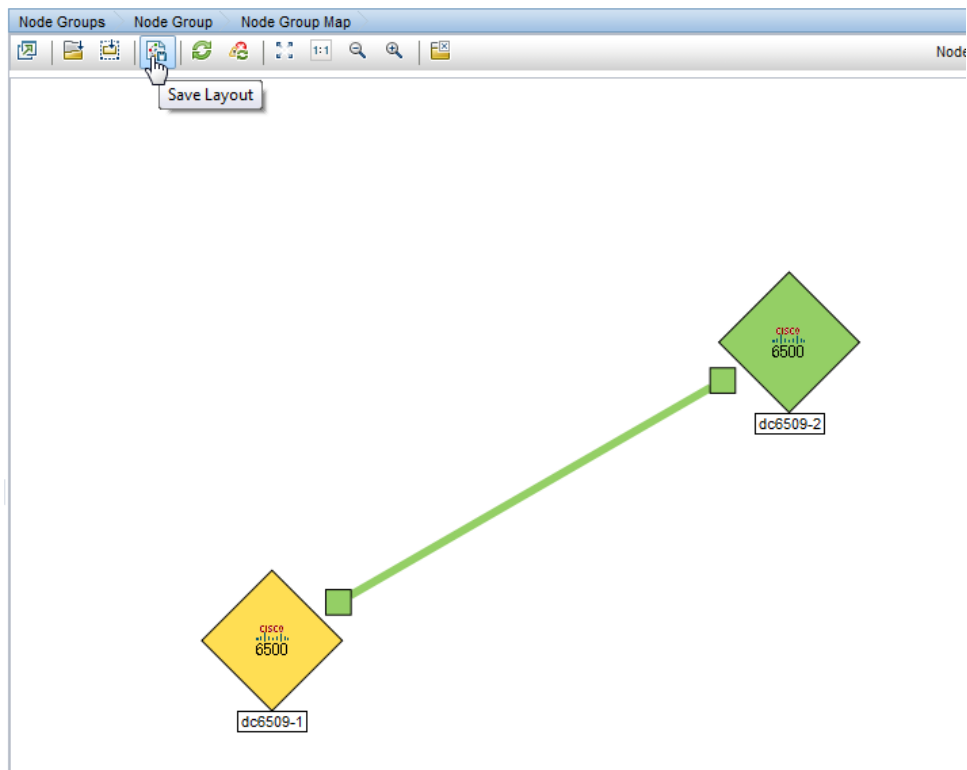
After you save the change, NNMi displays a message informing you that it created a Node Group map.

Repeat this same process for the entire hierarchy. It may take time for status to fully propagate to the Node Groups.

Configure the Node Group Map

You now have a map hierarchy that you can navigate within. From the workspace navigation panel, select the **Topology Maps** workspace. If you do not see the newly created Node Group Maps, try refreshing the browser or signing out and back into NNMi.

Figure 52: My Network Topology Map
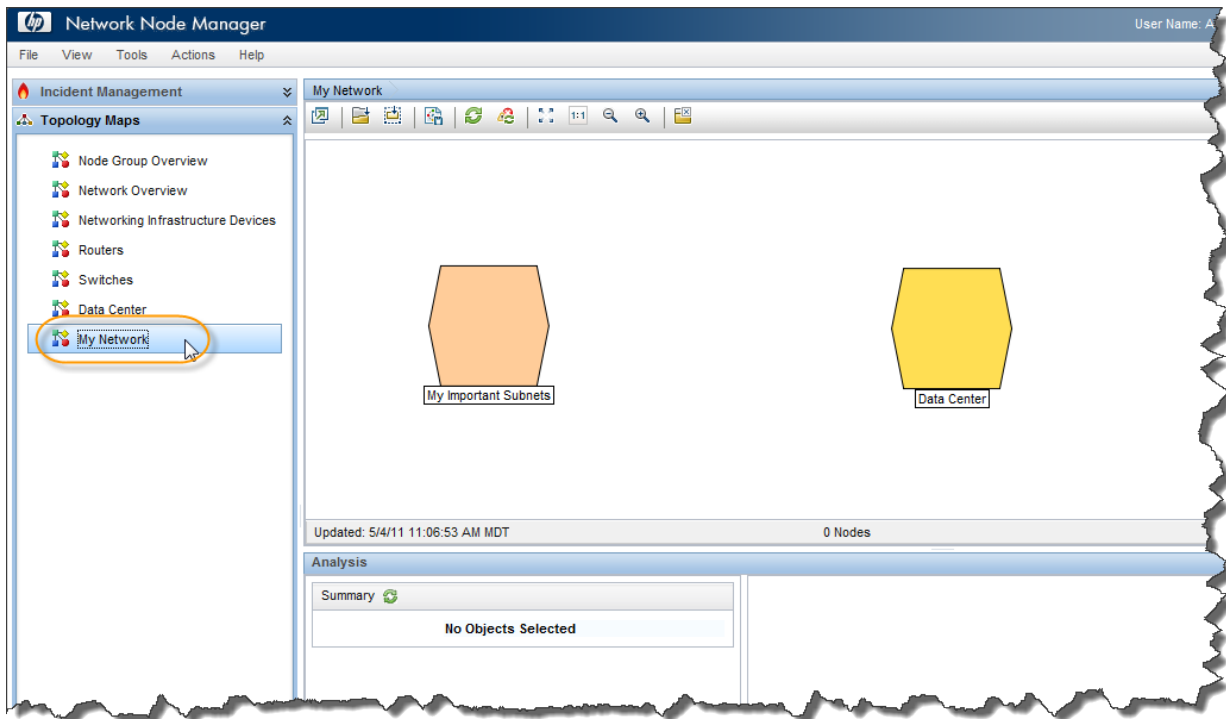
The bread crumb trail at the top indicates where you are in the hierarchy.

Figure 53: Bread Crumb Trail
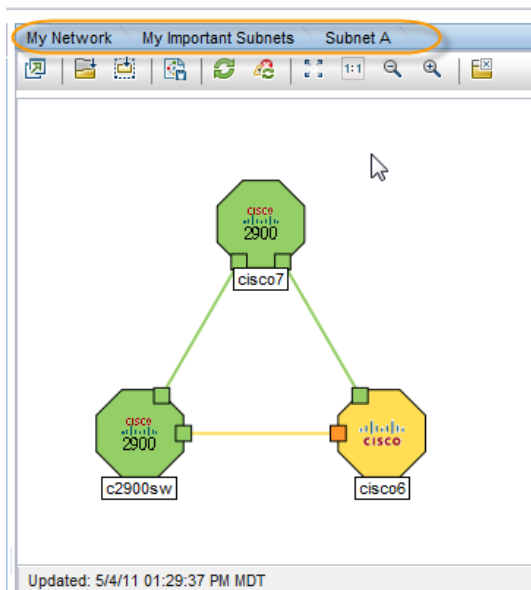
The Node Group Map Settings configuration option enables you to position Node Groups, add background graphics, and change connectivity options.

To place a background graphic on the map:

1. From the workspace navigation panel, select the **Configuration** workspace, expand the **User Interface** folder, and then click **Node Group Map Settings**.

Note the current Topology Map Ordering values. The lowest number currently used is 10.

Figure 54: Node Group Map Settings
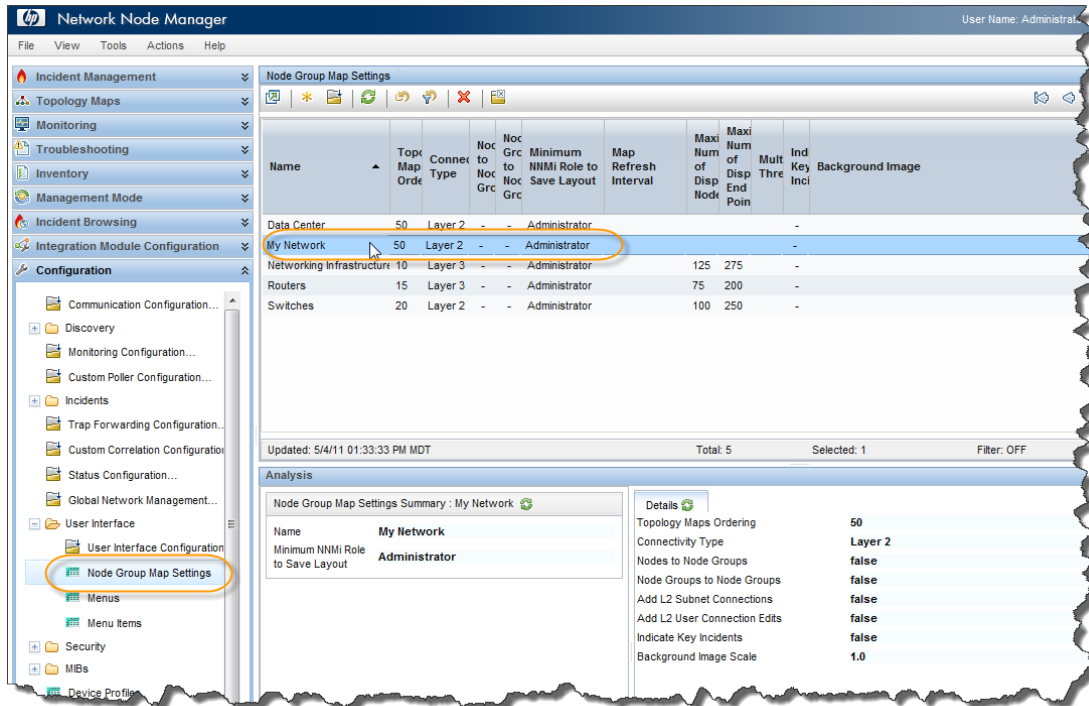
2. Double-click My Network.
3. Add a background image.

Tip: Use the local path, such as `/nnmbg/continents/europe.png`, rather than including `http://<machine name>` in front of the path. This enables the Application Failover feature to function properly.

4. Change the **Topology Maps Ordering** value to 5 so that this value is lower than the lowest value used in the previous example.
5. Click ⊞ **Save and Close**.

Figure 55: Save Node Group Map Settings
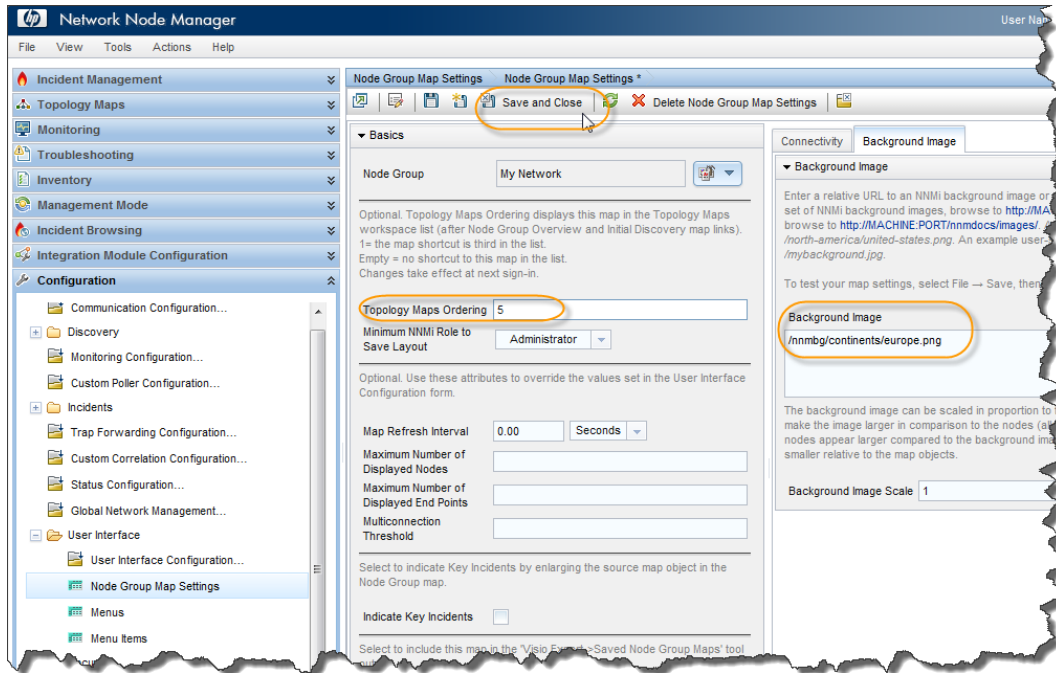
To specify the My Network map as the initial view:

6. Click **User Interface Configuration**.

Figure 56: Configuration: User Interface Configuration
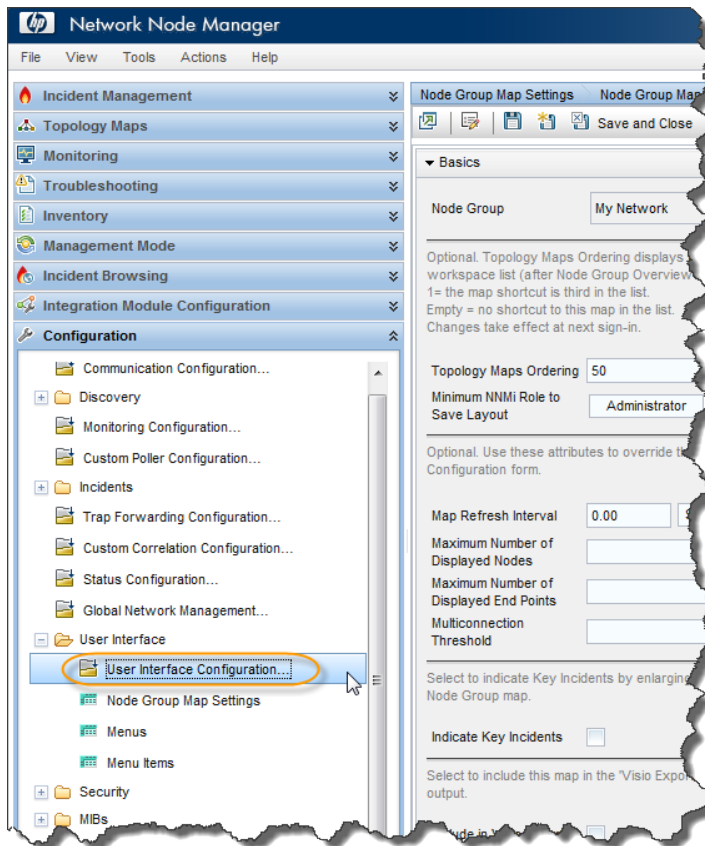
7. Change the **Initial View** selection to the **First Node Group in Topology Maps Workspace**. This is the My Network map because we set the **Topology Maps Ordering** attribute value to 5.

8. Click ⊠ **Save and Close**.

Figure 57: Save User Interface Configuration
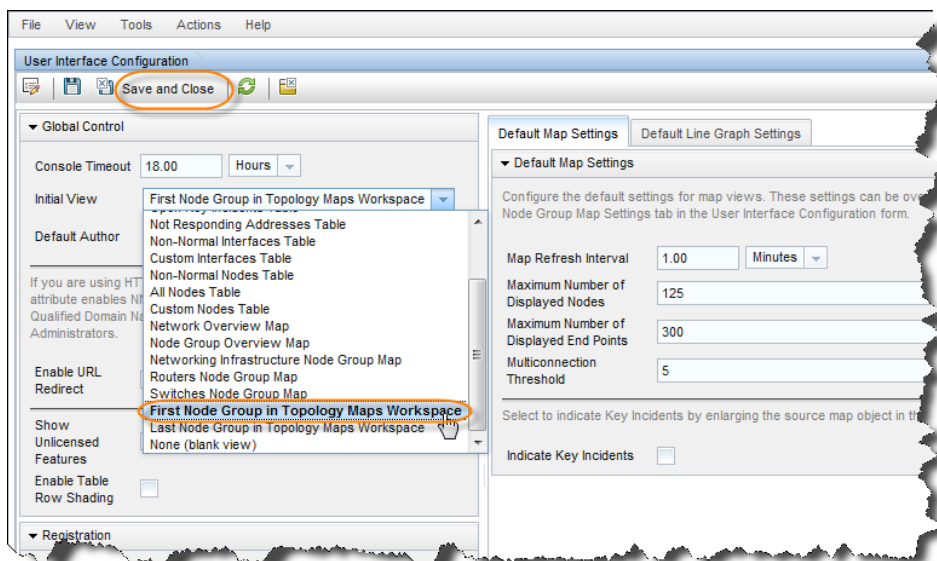
9. After you sign out, and then back into NNMi, the initial view is the My Network map.

Figure 58: My Network Map
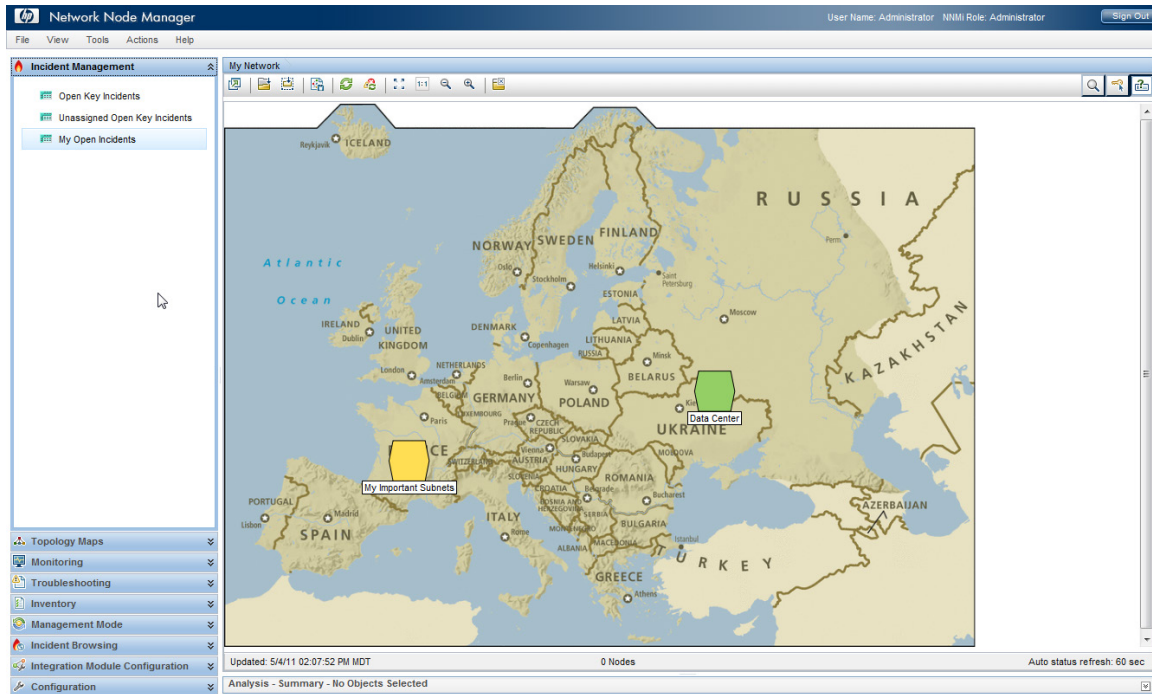
Maintain NNMi

Back up and Restore NNMi Data

NNMi provides backup and restore scripts to help protect your data.

The backup script is `nnmbackup.ovpl`. Use this script either *online* or *offline*. The online option enables you to run the script without stopping NNMi. Running this script generates a backup with a date and time stamp in the file name so you can specify the same target directory each time. This backup contains everything needed to restore your NNMi environment.

The following command shows an example of using the backup script:

```
nnmbackup.ovpl -type online -scope all -force -archive -target /var/tmp/mybackups
```

The previous command creates a file with a name similar to `nnm-bak-20110504145143.tar`.

The associated restore script is `nnmrestore.ovpl`. This command requires the backup file or directory created from the `nnmbackup.ovpl` script. To run this script, you must stop NNMi using the **ovstop -c** command.

An example `nnmrestore.ovpl` script usage is:

```
nnmrestore.ovpl -force -source /var/tmp/mybackups/nnm-bak-20110504145143.tar
```

The source directory should contain all of the files from the backup or the single tar file. If the source is a tar file, the script extracts the tar file to a temporary folder in the current working directory. The script removes the temporary folder after it completes the restore.

Caution: Never restore a backup across NNMi patch versions or restore a backup from a previous patch level of NNMi.

For example, in the following scenario, you should not restore the backup from the NNMi management running patch 4 onto the patch 5 code. This will cause fatal errors for NNMi:

- Patch 4 is running on your NNMi management server.
- After you run a backup, you upgrade to patch 5.

Tip: Track the version of the patch you are running in the backups by using a naming convention for the directories. For example, name the backup directory `patch4`.

Export and Import NNMi Configurations

Configuring NNMi is one of the most important tasks you do. Although your configuration is backed up as part of the `nnmbackup.ovpl` and `nnmbackupembdb.ovpl` scripts, consider using the `nnmconfigexport.ovpl` and `nnmconfigexport.ovpl` scripts included in NNMi. These scripts provide flexibility when it comes to restoring NNMi configuration. Using these scripts, you can:

- take a snapshot of the present NNMi configuration
- divide the configuration into small pieces
- restore just one piece of NNMi configuration if you need to revert back to a recent snapshot

For example, to create several Node Groups, use the export script to take a snapshot of the configuration at strategic points along the way so you can revert back if you make a significant mistake.

The export script is `nnmconfigexport.ovpl`. Use the `nnmconfigexport.ovpl` script to specify a configuration area, such as discovery, Node Groups, incidents, and many others. NNMi also provides an `all` option to export all of the configuration information.

See the `nnmconfigexport.ovpl` reference page or the UNIX manpage for details.

An example `nnmconfigexport.ovpl` script usage is listed below:

nnmconfigexport.ovpl –c nodegroup –f /tmp

In this example, NNMi displays the following message:

`Successfully exported /tmp/nodegroup.xml.`

Each exported configuration corresponds to one configuration area in the NNMi console.

Note: The `nnmconfigexport.ovpl` script does not generate a date and time stamp on the files. If you want to automate this command, put the date and time stamp in the directory name.

To restore the configuration, use the `nnmconfigimport.ovpl` script.

Tip: You do not need to specify a configuration area because this is implied by the file contents.

An example `nnmconfigexport.ovpl` script usage is listed below:

nnmconfigimport.ovpl –f /tmp/nodegroup.xml

As with the `nnmbackup.ovpl` and `nnmbackupembdb.ovpl` scripts, do not use these scripts across patch versions. NNMi validates the configuration file and rejects it during the import if it is invalid for the current version of NNMi.

Caution: The `nnmconfigimport.ovpl` script overrides the current configuration if the format is correct.

Note: NNMi does not support importing configurations from other NNMi management servers. Therefore, you cannot create a configuration export on one NNMi management server and import it on another server. Only a full backup (`nnmbackup.ovpl`) can be transferred between servers.

Trim Traps from the Database

Traps that pass all of the NNMi filters are eventually stored in the NNMi database. Traps can come in high volume and affect NNMi performance.

Tip: Regularly trim traps from your NNMi database using the `nnmtrimincidents.ovpl` script. You can archive these traps if necessary.

An example `nnmtrimincidents.ovpl` script usage is listed below:

`nnmtrimincidents.ovpl -age 1 -incr weeks -origin SnmpTrap -trimOnly -quiet`

This example usage trims any traps older than one week. This usage does not archive the traps. See the `nnmtrimincidents.ovpl` reference page or the UNIX manpage for more options.

Tip: Use `nnmtrimincidents.ovpl` in a cron job to clear out old unnecessary trap incidents on a regular basis.

Note: NNMi eventually forces you to trim traps from the NNMi database by stopping storage of traps after it reaches a limit of 100,000 traps in the NNMi database.

This reference to the NNMi database is not the same as the trap datastore. See the *Step- by-Step Guide to Incident Management,* available at http://h20230.www2.hp.com/selfsolve/manuals, for more information.

Check NNMi Health

You can check the general health of NNMi with a few different tools.

From the NNMi console, click **Help > System Information** for a listing of some important information.

Figure 59: Help: System Information
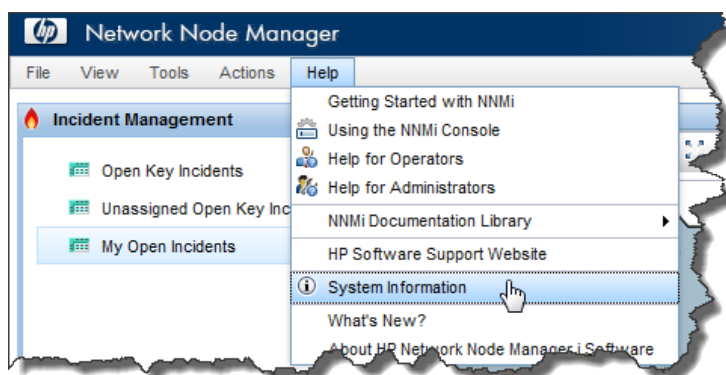
The best place to view the health of NNMi is in the **Health** tab. If NNMi identifies a health issue, it changes status and presents the reasons for the status in this report.

Figure 60: System Information: Health Tab
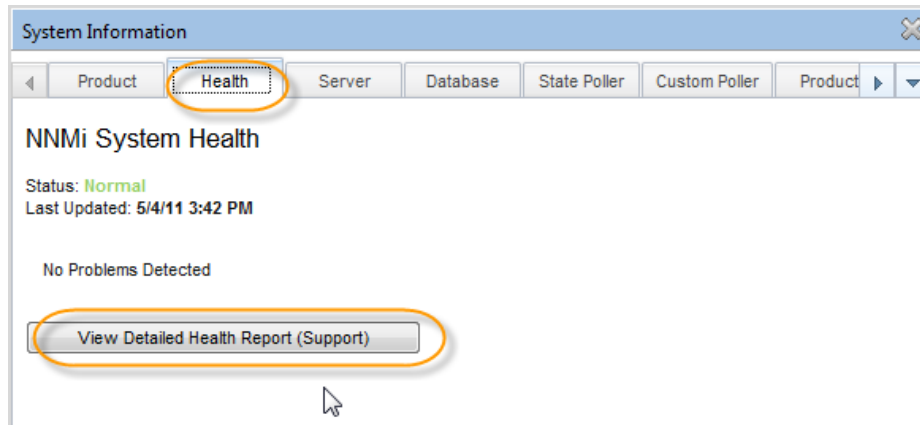
Best Practices

Some additional recommendations that you might want to consider:

- **NNMi Embedded Database**. Use NNMi's embedded database, even for large scale. Tests show that Postgres is highly scalable. You do not need to consider Oracle just because you have a large network. Postgres is highly reliable and is the preferred database for NNMi. Postgres is embedded into NNMi and NNMi provides any required tools you need.
- **SNMP Timeout Configuration**. Use caution when adjusting the SNMP timeout configuration. Timeout values increment with each timeout and can grow quickly beyond your original intention.
- **Node Status**. From the NNMi console, click one of the topology map selections. After you see the resulting display, double-click one of the nodes to open a node form. Click the **Conclusions** tab and review the data to better understand why the current status is set for the node.
- **Node Group Map Settings**. Reduce the number of connections between Node Groups using the `End Points Filter` in the **Node Group Map Settings** form. Highly connected maps display slowly and NNMi drops connections, if necessary, on the map.
- **SNMP Community Strings**. Do not use an @ symbol in your SNMP community strings. This is a reserved character for Cisco devices and causes unpredictable NNMi behavior.

Example Usage Scenarios

This section presents three usage scenarios. These scenarios assume that you have only NNMi available.

Management by Exception

NNMi identifies root cause problems associated with a network fault as Key Incidents.

To view all of the Open Key Incidents:

1. From the workspace navigation panel, select the **Incident Management** workspace.
2. Click **Open Key Incidents**.

NNMi displays all of the outstanding key incidents in your network and updates this list every 30 seconds. See "Help for Operators" in the NNMi help for more information about key incidents.

Tip: NNMi filters the Open Key Incidents view by time. Use the drop-down menu to select an appropriate time value.

The following example displays all of the open key incidents that occurred in the last day. Using this example, you can see that one node went down in the last 24 hours.

Figure 61: Open Key Incidents

By monitoring the Open Key Incidents view, you can pinpoint the exact cause of a network problem and begin working toward a solution. This is *management by exception* because the incident view shows these *exceptions* (or outages).

The *management by exception* approach includes the following advantages:

- You can quickly see the root cause of the problem.

- You can easily identify the source of the problem as the *source object*, such as an interface, address, node, or other possible sources.

- NNMi can forward Key Incidents to other products, such as HP Operations Manager (HP OM).

Note the following when using the *management by exception* approach:

- A Node Down incident shows only the root cause; however, the node being down could affect connectivity to many other nodes. Check the **Topology Maps** views to assist you in recognizing the scope of an outage. (See the following section, *Map-Based Management*, for more information.)

- Not all Node Down incidents are of equal importance. You will want additional tools, such as the **Topology Maps** view and Node Group names, to assist you in prioritizing these incidents. (See the following section, *Map-Based Management*, for more information.)

Map-Based Management

Another method of network management is to create maps to monitor node status changes. These maps can be arranged in many ways, including geography or building.

All of the maps available from the **Topology Maps** workspace are arranged by Node Groups. Note the following about Node Group maps:

- The status is propagated from the *Child Node Group nodes* up to the parent Node Group maps.
- By default, NNMi propagates the most critical node status in the Node Group up the hierarchy. This enables you to monitor node status from a high level.
- When a top-level Node Group map changes color from green to red, yellow, or orange, you can navigate into the Node Group maps until you find the problem node. After you reach the problem node, you can take actions similar to those described in the previous section to troubleshoot the problem.
- Similar to incidents, nodes and interfaces can be annotated with notes if you want to keep a log of information about the troubleshooting progress.

The following screen capture shows an example of the My Network map with a problem that you need to correct. In this example, double-click the Node Group icon to find the faulting node.

Tip: The NNMi administrator can specify the *default map* that NNMi displays after initial sign in.

To navigate to a Node Group map from the NNMi console, click **Topology Maps**, and then select the map name of interest.

Figure 62: My Network Topology Map
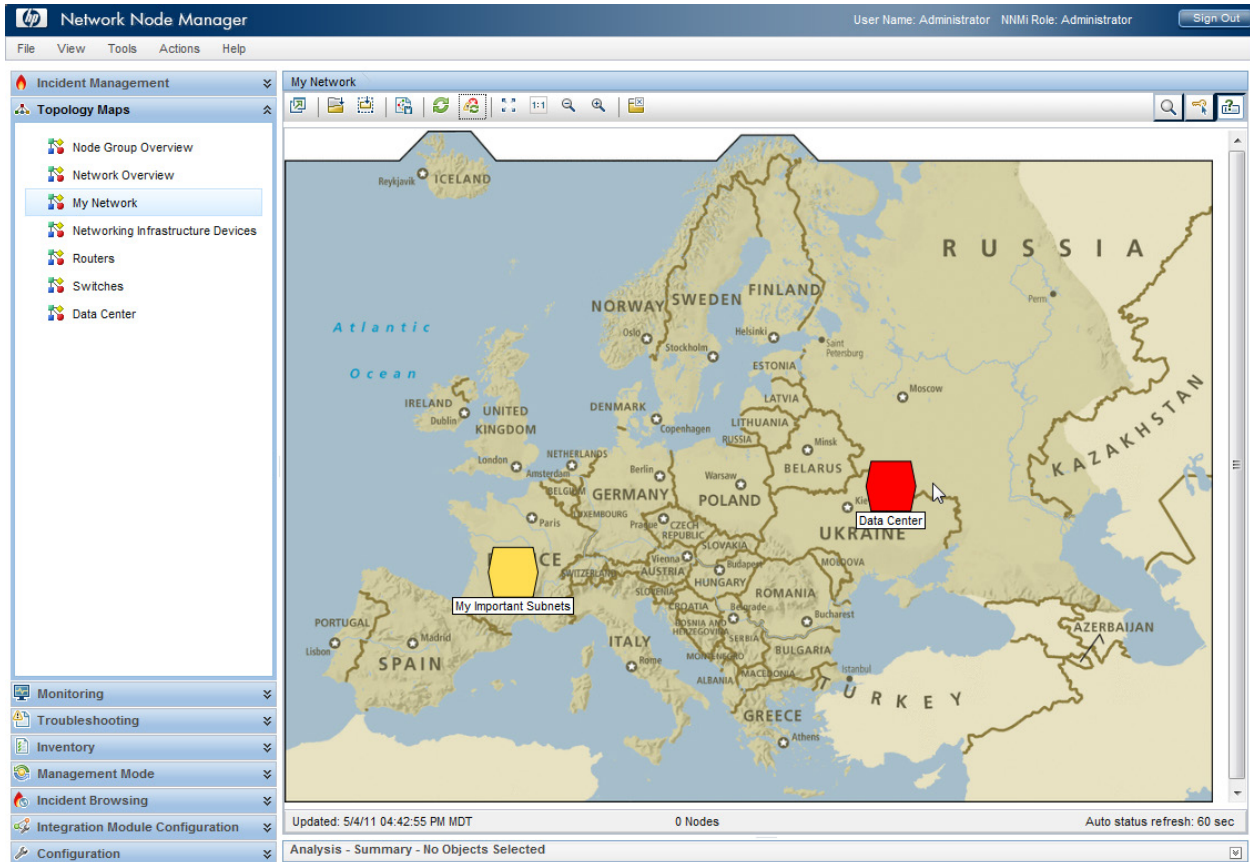
The *map-based management* approach includes the following advantages:

- You can easily scope the outage. It becomes obvious quickly if other nodes are affected based on the status of neighboring nodes.
- You can easily identify the affected location. This approach helps you decide what to work on first.

When using the *map-based management* approach note the following:

- To find the source of the problem, open the node and go to the **Conclusions** tab to determine the problem.
- If one node is already down in a Node Group, NNMi does not indicate that one or more additional nodes have gone down in the same Node Group.
- NNMi does not propagate node status to other tools such as HP Operations Manager (HP OM).

List-Based Management

NNMi also enables you to manage your network from a dynamic list. NNMi provides dynamically updated tables that show nodes or interfaces experiencing problems. NNMi usually updates this list every 15 seconds. From this list, you can use tools, as described in the previous sections, to diagnose and fix problems. Because this list is dynamic, NNMi removes the nodes or interfaces from this list as the nodes or interfaces return to a Normal status.

For example, to display all the nodes having a non-normal status:

1. From the workspace navigation panel, select the **Monitoring** workspace.
2. Click **Non-Normal Nodes**.

As shown in the following example, NNMi displays all nodes that have a status other than Normal.

Figure 63: Non-Normal Nodes

The *list-based management* approach includes the following advantages:

- You know how many nodes or interfaces you need to investigate.
- You do not need to navigate into NNMi maps to troubleshoot your network.

When using list-based management, note the following:

- NNMi includes up to five entries in the status history.
- NNMi does not assign a Critical status to nodes that are "in the shadow" of a node that is down. See "Help for Operators" in the NNMi help for more information.
- The list-based view does not indicate where the node is physically located.
- NNMi does not propagate node status to other tools such as HP Operations Manager (HP OM).

Conclusion

This document described an NNMi deployment on a small test network. It included information about installing a license, creating users, configuring communication, discovery, incidents, traps, actions, and the NNMi console. This document also explained maintenance tasks for NNMi and how to monitor NNMi health. It also provided some best practices and explained some possible usage scenarios for NNMi.

Legal Notices

Support

Visit the HP Software Support web site at:

www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp