

# HP Network Node Manager i Software

For the Windows<sup>®</sup>, HP-UX, Linux, and Solaris operating systems

Software Version: 9.20

---

## Upgrade Reference

Document Release Date: May 2012  
Software Release Date: May 2012



## Legal Notices

### Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

### Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

### Copyright Notice

© Copyright 2008–2012 Hewlett-Packard Development Company, L.P.

### Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

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

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

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

### Oracle Technology — Notice of Restricted Rights

Programs delivered subject to the DOD FAR Supplement are 'commercial computer software' and use, duplication, and disclosure of the programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, programs delivered subject to the Federal Acquisition Regulations are 'restricted computer software' and use, duplication, and disclosure of the programs, including documentation, shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software-Restricted Rights (June 1987). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

For the full Oracle license text, see the license-agreements directory on the NNMi product DVD.

### Acknowledgements

This product includes software developed by the Apache Software Foundation.  
(<http://www.apache.org>)

This product includes software developed by the Indiana University Extreme! Lab.  
(<http://www.extreme.indiana.edu>)

## Available Product Documentation

In addition to this guide, the following documentation is available for NNMi:

- *HP Network Node Manager i Software Documentation List*—Available on the HP manuals web site. Use this file to track additions to and revisions within the NNMi documentation set for this version of NNMi. Click a link to access a document on the HP manuals web site.
- *HP Network Node Manager i Software Installation Guide*—Available for each supported operating system on the product media and the NNMi management server.
- *HP Network Node Manager i Software Deployment Reference*—Available on the HP manuals web site.
- *HP Network Node Manager i Software Release Notes*—Available on the product media and the NNMi management server.
- *HP Network Node Manager i Software System and Device Support Matrix*—Available on the product media and the NNMi management server.
- *HP Network Node Manager iSPI Network Engineering Toolset Planning and Installation Guide*—Available on the NNM iSPI NET diagnostics server product media.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

**<http://h20230.www2.hp.com/selfsolve/manuals>**

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

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

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

## Support

Visit the HP Software Support Online web site at:

**[www.hp.com/go/hpsoftwaresupport](http://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 user 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](http://h20230.www2.hp.com/new_access_levels.jsp)**

# Contents

<b>About This Guide</b> .....	<b>9</b>
What Is in This Guide? .....	9
Path Conventions Used in This Document .....	11
Revision History .....	11
<b>Upgrading from 6.x or 7.x</b> .....	<b>13</b>
<b>Upgrading from 6.x or 7.x</b> .....	<b>15</b>
Upgrade Options .....	15
Start with a New Installation .....	15
Upgrading in Phases .....	16
Phase 1: Collect Data from the NNM Management Station .....	19
Phase 2: Upgrade SNMP Information .....	21
Configure SNMP Access .....	21
Limit Name Resolution .....	25
Customize Device Profiles .....	26
Phase 3: Upgrade Discovery .....	27
Schedule Discovery .....	28
Select Your Discovery Method .....	29
Configure Auto-Discovery Rules .....	30
Configure Spiral Discovery .....	30
Exclude Addresses from Discovery .....	34
Add Seeds to NNMi for Seeded Discovery .....	35
Customize Connectivity .....	36
Phase 4: Upgrade Status Monitoring .....	37
Set Polling Intervals .....	37
Select Polling Protocol .....	39
Configure Critical Nodes .....	41
Exclude Objects from Status Polling .....	42
Phase 5: Upgrade Event Configuration and Event Reduction .....	43
Display Traps from Devices .....	43
Customize Display of NNMi-Generated Management Events .....	45
Block/Ignore/Disable Traps .....	45
Configure Lifecycle Transition Actions .....	46
Configure Additional (Manual) Actions .....	47
Event Correlation: Repeating Events .....	48
Event Correlation: Counting the Rate .....	49
Event Correlation: Pairwise Cancellation .....	50
Event Correlation: Scheduled Maintenance .....	50

Phase 6: Upgrade Graphical Visualization (OVW) .....	50
Phase 6: Upgrade Graphical Visualization (Home Base) .....	53
Phase 7: Upgrade Custom Scripts .....	53
Upgrade Tools Reference .....	54
Data Collection Tools .....	54
NNM Configuration Data Files .....	56
Data Import Tools for Upgrading .....	57
<b>Integrating NNM 6.x or NNM 7.x with NNMi .....</b>	<b>59</b>
Configure Event Forwarding .....	60
Step 1: Configure NNM 6.x or 7.x to Forward Events to the NNMi Management Server .....	60
Recommended and Supported Procedure: Use the Event Configuration Window .....	60
Optional: Destination List File .....	61
Alternative Procedure: Manually Edit trapd.conf .....	62
Step 2: (Optional) Use Node Level Filtering to Further Reduce Events .....	62
Step 3: Add the NNM 6.x or 7.x Management Station to the NNMi Topology .....	62
Step 4: (Optional) Save the Management Station Configuration .....	63
Step 5: Verify NNM 6.x or 7.x Incident Configuration in the NNMi Console .....	63
Mapping Categories .....	63
Configure Remote View Launching .....	64
Step 1: Install Java Plug-in .....	64
Step 2: Create an NNM 6.x or 7.x Management Station Entity in NNMi .....	65
Step 3: (Optional) Configure Additional NNM 6.x or 7.x Views .....	66
URLs That Do Not Require a Selection .....	66
URLs That Require a Selection .....	67
Test the Integration .....	67
Test 1: Verify Event Forwarding .....	67
Generate Test Interface Down and Interface Up Events .....	68
sendMsg.ovpl .....	69
Test with Traps to NNM 6.x or 7.x System .....	69
Test 2: Launch NNM 6.x or 7.x Dynamic Views from NNMi .....	69
Troubleshoot Event Forwarding .....	70
<b>Upgrading from NNMi 8.0x or 8.1x .....</b>	<b>71</b>
Upgrading the NNMi Management Server in Place from 8.0x or 8.1x .....	73
Start from NNMi 8.0x .....	73
Upgrade an Existing NNMi Management Server to NNMi 9.0x .....	73
Upgrading to a Different NNMi Management Server from 8.0x or 8.1x .....	75
Start from NNMi 8.0x .....	75
Upgrade to a Different NNMi Management Server .....	75
Changing the NNMi Management Server from 8.0x or 8.1x .....	77
Best Practices for Preparing the NNMi Configuration to Be Moved .....	77
Moving the NNMi Configuration and Embedded Database .....	78
Moving the NNMi Configuration .....	79
Restoring the NNMi Public Key Certificate .....	79
Changing the IP Address of a Standalone NNMi Management Server .....	82

Changing the Hostname or Domain Name of an NNMi Management Server . . . . .	83
Changing the Oracle Database Instance Connection Information . . . . .	86
Changing the Password that NNMi Uses to Connect to the Oracle Database Instance . . . . .	87
<b>Moving NNMi from Red Hat Linux 4.6 to 5.2 or 5.3 . . . . .</b>	<b>89</b>
Changing NNMi from Red Hat Linux 4.6 to Red Hat Linux 5.2 or 5.3 . . . . .	89
<b>Migrating NNMi Oracle Data . . . . .</b>	<b>93</b>
Migrating NNMi Oracle Data . . . . .	93
<b>Additional Upgrade Information . . . . .</b>	<b>95</b>
Configuration Differences . . . . .	95
Functionality Differences . . . . .	96
<b>Application Failover and Upgrading from NNMi 8.x to NNMi 9.0x. . . . .</b>	<b>99</b>
Application Failover and Upgrading to NNMi 9.00 . . . . .	99
Application Failover and NNMi Patches . . . . .	101
<b>High Availability and Upgrading from NNMi 8.1x to NNMi 9.0x . . . . .</b>	<b>103</b>
Upgrading NNMi under HA from NNMi 8.1x to NNMi 9.01 . . . . .	103
Unconfiguring NNMi from an HA Cluster . . . . .	106
Unconfiguring NNMi from an HA Cluster . . . . .	106
Running NNMi with the Existing Database Outside HA . . . . .	109
Patching NNMi under HA . . . . .	110
<b>Upgrading from NNMi 9.0x or 9.1x . . . . .</b>	<b>113</b>
<b>Upgrading the NNMi Management Server in Place. . . . .</b>	<b>117</b>
Upgrade an Existing NNMi Management Server to NNMi 9.20 . . . . .	117
<b>Upgrading to a Different NNMi Management Server . . . . .</b>	<b>119</b>
Upgrade to a Different NNMi Management Server. . . . .	119
<b>Moving NNMi from Windows 2003 to Windows 2008. . . . .</b>	<b>121</b>
Changing NNMi from Windows 2003 to Windows 2008 . . . . .	121
<b>Moving NNMi from a RHEL Version below 5.4 to RHEL 5.4 or Greater. . . . .</b>	<b>125</b>
Changing NNMi from RHEL (Versions below 5.4) to RHEL Version 5.4 or Greater. . . . .	125
<b>Migrating NNMi Oracle Data . . . . .</b>	<b>129</b>
Migrating NNMi Oracle Data . . . . .	129
<b>Additional Upgrade Information . . . . .</b>	<b>131</b>
Configuration Differences . . . . .	131
Application Failover . . . . .	132
MIBs . . . . .	133
Functionality Differences . . . . .	133
<b>Upgrading Global and Regional Managers from NNMi 9.0x or 9.1x. . . . .</b>	<b>135</b>
NNMi Versions Supported by Global Network Management . . . . .	135
Global Network Management Upgrade Steps . . . . .	135

**Application Failover and Upgrading to NNMi 9.20 . . . . . 137**

- Application Failover and Upgrading from NNMi 9.0x or 9.1x . . . . . 137
  - Embedded Database . . . . . 137
  - Oracle Database . . . . . 140
- Application Failover and NNMi Patches . . . . . 142
  - Applying Patches for Application Failover (Shut Down Both Active and Standby) . . . . . 142
  - Applying Patches for Application Failover (Keep One Active NNMi Management Server) . . . . . 144

**High Availability and Upgrading from NNMi 9.0x or 9.1x to NNMi 9.20 . . . . . 147**

- Upgrade NNMi with the Embedded Database on all Supported Operating Systems . . . . . 147
- Upgrade NNMi with Oracle on all Supported Operating Systems . . . . . 151
- Unconfiguring NNMi from an HA Cluster . . . . . 151
  - Running NNMi Outside HA with the Existing Database . . . . . 154
- Patching NNMi under HA . . . . . 155

**We appreciate your feedback! . . . . . 157**



# About This Guide

This chapter contains the following topics:

- [What Is in This Guide?](#)
- [Path Conventions Used in This Document](#)
- [Revision History](#)

---

## What Is in This Guide?

This guide contains information for upgrading from the following HP Network Node Manager (NNM) and HP Network Node Manager i Software (NNMi) versions to NNMi 9.20:

- [Upgrading from 6.x or 7.x on page 13](#)
- [Upgrading from NNMi 8.0x or 8.1x on page 71](#)
- [Upgrading from NNMi 9.0x or 9.1x on page 113](#)

Also see *NNMi 9.20 Upgrade Path Requirements* available at: <http://h20230.www2.hp.com/selfsolve/manuals>

This guide is for an expert system administrator, network engineer, or HP support engineer with experience deploying and managing networks in large installations.

Note the following product naming conventions:

- **NNM** refers to older versions of HP Network Node Manager (including all 6.x and 7.x releases of NNM).
- **NNMi** refers to HP Network Node Manager i Software (including all 8.x and all 9.x releases of NNMi and NNMi Advanced).

Before using this guide, make sure you have completed the following tasks:

- You have installed the version of NNM or NNMi from which you are upgrading using either of the following:
  - *HP Network Node Manager Installation Guide*

- *HP Network Node Manager i Software Installation Guide*
- *HP Network Node Manager i Software Interactive Installation Guide*
- You have reviewed the concepts described in the NNMi help and the deployment information in the *NNMi Deployment Reference* for a general understanding of NNMi functions.
- You understand how to use the NNMi console.

For up-to-date, downloadable copies of NNM and NNMi documentation, go to:

**<http://h20230.www2.hp.com/selfsolve/manuals>**

The information in this guide was formerly published in the *NNMi Deployment Reference*.

HP updates this guide between product releases, as new information becomes available. For information about retrieving an updated version of this document, see [Available Product Documentation](#) on page 3.

## Path Conventions Used in This Document

For commands located in the NNMi bin directory, this document does not include the command path. The NNMi bin directory is located as follows:

- *Windows Server 2008*: <drive>\Program Files\HP\HP BTO Software\bin
- *UNIX*<sup>®</sup>: /opt/OV/bin

This document primarily uses the following two NNMi environment variables to reference file and directory locations. This list shows the default values. Actual values depend on the selections that you made during NNMi installation.

- *Windows Server 2008*:
  - %NnmInstallDir%: <drive>\Program Files\HP\HP BTO Software
  - %NnmDataDir%: <drive>\ProgramData\HP\HP BTO Software



On Windows systems, the NNMi installation process creates these system environment variables, so they are always available to all users.

- *UNIX*:
  - \$NnmInstallDir: /opt/OV
  - \$NnmDataDir: /var/opt/OV



On UNIX systems, you must manually create these environment variables if you want to use them.

Additionally, this document references some of the NNMi environment variables that you can source as part of your user log-on configuration on the NNMi management server. These variables are of the form `NNM_*`. For information about this extended list of NNMi environment variables, see “Other Available Environment Variables” in the *NNMi Deployment Reference*.


## Revision History


The following table lists the major changes for each new release of this document.

Document Release Date	Description of Major Changes
March 2011 (9.10)	Entirely updated. <ul style="list-style-type: none"> <li>• Second English edition.</li> <li>• First Japanese edition. (Content was previously published in the <i>NNMi Deployment Reference</i>)</li> </ul>
May 2012 (9.20)	<ul style="list-style-type: none"> <li>• Included upgrade information from multiple versions into one consolidated manual.</li> <li>• Added new section on <a href="#">Upgrading from NNMi 9.0x or 9.1x</a>.</li> </ul>



# Upgrading from 6.x or 7.x

 For information about upgrading from NNMi 8.1x to NNMi 9.20, see [Upgrading from NNMi 8.0x or 8.1x](#) on page 71.

 For information about upgrading from NNMi 9.0x/9.1x to NNMi 9.20, see [Upgrading from NNMi 9.0x or 9.1x](#) on page 113.

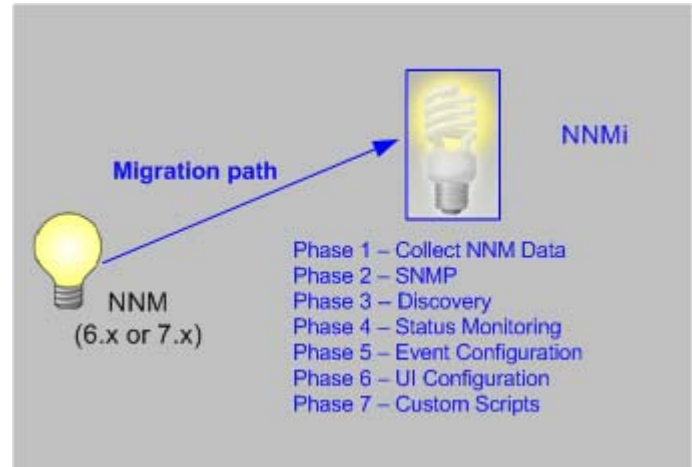
This section provides a basic path for upgrading from HP Network Node Manager (NNM) 6.x or 7.x to NNMi 9.20. This chapter does not cover advanced upgrade topics or customizations; consulting services are available to meet your needs in these areas.

To complete an upgrade from NNM 6.x/7.x to NNMi 9.20, perform the following tasks:

- [Upgrading from 6.x or 7.x](#)
- [Integrating NNM 6.x or NNM 7.x with NNMi](#)



# Upgrading from 6.x or 7.x



This chapter contains the following topics:

Upgrade Options

Phase 1: Collect Data from the NNM Management Station

Phase 2: Upgrade SNMP Information

Phase 3: Upgrade Discovery

Phase 4: Upgrade Status Monitoring

Phase 5: Upgrade Event Configuration and Event Reduction

Phase 6: Upgrade Graphical Visualization (OVW)

Phase 6: Upgrade Graphical Visualization (Home Base)

Phase 7: Upgrade Custom Scripts

Upgrade Tools Reference

---

## Upgrade Options

### Start with a New Installation

If your NNM installation is more than two years old, consider using this opportunity to begin with a new installation. Completely re-evaluating how to manage your current network might result in a significant overhead drop and a streamlined operation compared with your NNM environment.

If you choose to start with a new installation of NNMI, install NNMI by following the instructions in the *HP Network Node Manager i Software Interactive Installation Guide*. Then consider the deployment tasks presented in the *NNMi Deployment Reference*. You do not need to read this chapter.

## Upgrading in Phases

For some organizations, a phased approach to upgrading works better than a new installation. These organizations require that the new NNMi implementation completely reproduce and replace the existing NNM implementation. While there are many possible paths to that end, HP recommends the following phases:

- **Phase 1: Collect Data from the NNM Management Station**

Use the NNMi-provided tools to gather the information needed for upgrading from the NNM management station.

- **Phase 2: Upgrade SNMP Information**

Configure NNMi with the SNMP access information for your environment.

- **Phase 3: Upgrade Discovery**

Configure NNMi to discover the objects that were discovered by NNM by approximating the way that NNM discovered them (automatically).

- **Phase 4: Upgrade Status Monitoring**

Configure the status polling intervals and protocols that are most appropriate for your environment.

- **Phase 5: Upgrade Event Configuration and Event Reduction**

Configure NNMi to display the event severity, category, message, and to perform the automatic actions you had configured in NNM. You might also need to configure deduplication, rate counting, pairwise cancellation, and threshold monitoring.



- Phase 6: Upgrade Graphical Visualization  
Select one of the following approaches:
  - Phase 6: Upgrade Graphical Visualization (OVW)  
Configure NNMi with node group maps that are similar to the NNM OVW location submaps.
  - Phase 6: Upgrade Graphical Visualization (Home Base)  
Configure NNMi with node group maps that are similar to the NNM 7.x Advanced Edition Home Base container views.
- Phase 7: Upgrade Custom Scripts  
Update scripts that use NNM command line tools to call NNMi command line tools.



NNMi can act as a manager of managers for your existing NNM systems. You can configure NNM to forward events to NNMi. Then you can use the NNMi console, with its consolidated user interface, incident ownership, and lifecycle states to navigate to familiar NNM tools. For instructions on integrating NNM into NNMi, see [Integrating NNM 6.x or NNM 7.x with NNMi](#) on page 59.

**Table 1** presents a high-level overview of the upgrade process for the two ends of the upgrade complexity continuum:

- The simplest approach involves importing environment-specific information from NNM and accepting the default NNMi configuration values, which are improved from NNM.
- The most detailed and thorough approach takes a close look at the NNM configuration and replicates this configuration in NNMi.

The remainder of this chapter walks through the process of replicating an NNM configuration in NNMi. The text in the left margin indicates how the specific steps fit into the upgrade process:

- **Gather from NNM** indicates work to be done on the NNM management station.
- **Replicate to NNMi** indicates work to be done on the NNMi management server.
- **Enhance in NNMi** indicates optional work to do on the NNMi management server. You can perform enhancements during the upgrade process or at any time in the future.

At appropriate points, you will be given two or more options along the complexity continuum for completing a given task.

**Table 1 Upgrade Continuum**

Phase	Simplest Approach	Most Detailed and Thorough Approach
<b>Collect Data from NNM</b>	<ol style="list-style-type: none"> <li>1 Use the NNMi-provided tools on the NNM management station.</li> <li>2 Copy the collected data to the NNMi management server.</li> </ol>	<ol style="list-style-type: none"> <li>1 At each upgrade phase, gather the appropriate NNM configuration data by hand.</li> <li>2 Copy the collected data to the NNMi management server.</li> </ol>
<b>SNMP Information</b>	Import the collected community strings into NNMi, and let NNMi sort out which community string goes with which node.	<ol style="list-style-type: none"> <li>1 Export all community strings currently in use.</li> <li>2 Modify the data file and import the contents to NNMi as specific node community strings.</li> </ol>
<b>Discovery</b>	Modify the collected list of discovered nodes, and import the file contents into NNMi as seeds with no auto-discovery rules.	<ol style="list-style-type: none"> <li>1 Determine how NNM and <code>netmon</code> find nodes (seeds, loadhosts, filters, other tools).</li> <li>2 Replicate this approach as closely as possible with seeds and auto-discovery rules.</li> </ol>
<b>Status Monitoring</b>	NNMi defaults are updated to match most customer requirements. You might not need to make significant changes to these default values, so begin with the updated default values.	<ol style="list-style-type: none"> <li>1 Determine exactly what polling intervals and polling policies were used by NNM and <code>netmon</code> or APA for each group of nodes.</li> <li>2 Implement NNMi node groups and interface groups to replicate the polling intervals and polling policies.</li> </ol>
<b>Event Configuration and Event Reduction</b>	<ol style="list-style-type: none"> <li>1 Start with the default configuration from NNM.</li> <li>2 Add the definitions for any custom traps from managed devices.</li> <li>3 Add automatic actions as necessary.</li> </ol>	<ol style="list-style-type: none"> <li>1 Determine exactly what NNM customizations have been made for each trap and event type.</li> <li>2 Customize each matching trap and event type on the NNMi system.</li> </ol>
<b>Graphical Visualization</b>	<ol style="list-style-type: none"> <li>1 Import the NNM <code>ovw</code> containers.</li> <li>2 Assign node groups to containers.</li> </ol> <p><b>OR</b></p> <ol style="list-style-type: none"> <li>1 Import the NNM 7.x Advanced Edition container views.</li> <li>2 Assign node groups to containers.</li> </ol>	<ol style="list-style-type: none"> <li>1 In the most inclusive NNM map, determine what is on each submap.</li> <li>2 Create a node group for the contents of each NNM submap.</li> <li>3 For each node group, create an NNMi map, add a background image, and place each node.</li> </ol>
<b>Custom Scripts</b>	Modify existing scripts to use the <code>nnmtopodump.ovpl</code> command.	Write new scripts that incorporate the new tools in NNMi.

## Phase 1: Collect Data from the NNM Management Station

NNMi provides tools that run on the NNM management station to collect the majority of data needed for replicating the NNM configuration to NNMi. The tools create text files from information in the NNM databases and copy other configuration information. The tools also assemble the data into a known directory structure for copying to the NNMi management server.

For information about the data collection tools and the information that these tools collect, see [Data Collection Tools](#) on page 54.

### Gather from NNM

#### Upgrade tool approach

- 1 Perform a complete back up of the NNM system.
- 2 Copy the data collection tool archive from the NNMi management server to the NNM management station. The file name and locations depend on the operating system of each computer.
  - On the NNMi management server, the archive is in the following directory:
    - *Windows*: %NnmInstallDir%\migration\
    - *UNIX*: \$NnmInstallDir/migration/
  - On the NNM management station, place the archive as follows:
    - *Windows*: Copy the migration.zip file to the NNM installation folder (*install\_dir*, usually similar to C:\Program Files\HP OpenView).
    - *UNIX*: Copy the migration.tar file to the /opt/OV/ directory.
- 3 Unpack the data collection tool archive using a tool or command that is appropriate for the operating system of the NNM management station.
- 4 Set localization environment variables for your environment.
- 5 From the NNM installation directory, run the tools:
  - a Change to the migration directory.
  - b Create the expected directory structure for the data to be collected:
 

```
bin/createMigrationDirs.ovpl
```
  - c Collect the NNM data:
 

```
bin/nmmigration.ovpl
```
  - d If you want to include the OVW map location hierarchy data in the upgrade archive, complete the upgrade tool approach for gathering the map data as described in [Phase 6: Upgrade Graphical Visualization \(OVW\)](#) on page 50.



If Home Base container views are configured on the NNM management station, this information is included in the upgrade archive. No additional work is necessary.

- e Archive the collected data:

**bin/archiveMigration.ovpl**

This tool creates the following file:

- *Windows*: %NnmDataDir%\tmp\migration\*<hostname>*.tar
- *UNIX*: \$NnmDataDir/tmp/migration/*<hostname>*.tar

The *<hostname>*.tar file contains the collected data for simple data transfer to the NNMi management server. The tool consumes a large amount of memory while it is running. If the NNM system does not have enough available memory or disk space, this tool fails; you can archive the data yourself in smaller chunks or copy individual files as needed.



On Windows operating systems, archiveMigration.ovpl might run slowly. Consider using another tool for archiving the data in preparation for moving it to the NNMi system.

### Manual approach

If the upgrade tool approach does not work in your environment, follow the steps listed in each phase for gathering NNM data at that time.

### Replicate to NNMi

Copy the data archive to the NNMi management server.

### Upgrade tool approach

If the archiveMigration.ovpl tool completed successfully, follow these steps:

- 1 On the NNMi management server, change to the following directory:
  - *Windows*: %NnmDataDir%\tmp\*<hostname>*
  - *UNIX*: \$NnmDataDir/tmp/*<hostname>*
- 2 In the tmp directory, create the migration and *<hostname>* directories in the following structure:
  - *Windows*: %NnmDataDir%\tmp\migration\*<hostname>*
  - *UNIX*: \$NnmDataDir/tmp/migration/*<hostname>*
- 3 Copy the *<hostname>*.tar file from the NNM management station to the following location on the NNMi management server:
  - *Windows*: %NnmDataDir%\tmp\migration\*<hostname>*\*<hostname>*.tar
  - *UNIX*: \$NnmDataDir/tmp/migration/*<hostname>*/*<hostname>*.tar
- 4 On the NNMi management server, change to the directory that you created in step 2:
  - *Windows*: %NnmDataDir%\tmp\migration\*<hostname>*
  - *UNIX*: \$NnmDataDir/tmp/migration/*<hostname>*
- 5 Unpack the data archive:
  - *Windows*:
 

```
%NnmInstallDir%\migration\bin\restoreMigration.ovpl \
-source <hostname>.tar
```
  - *UNIX*:

```
$NnmInstallDir/migration/bin/restoreMigration.ovpl \  
-source <hostname>.tar
```

### Manual approach

If the `archiveMigration.ovpl` command did not complete successfully, copy the data files manually.



The process of copying a text file from Windows to UNIX can insert `^M` characters into the file.

- To avoid this problem, transfer files using FTP in ASCII mode.
- To remove `^M` characters from a text file, on the UNIX system run the `dos2ux` (or similar) command.

## Phase 2: Upgrade SNMP Information

Configure the SNMP community string information that NNMi uses to establish connections with managed devices.

If the NNM configuration includes IP addresses or hostnames that should not be looked up in the name resolution service, replicate that information in NNMi.

Customize NNMi device profiles for the custom devices in your network.

### Configure SNMP Access

NNMi discovery requires SNMP access to the managed nodes to collect specific information about their configuration and connectivity. SNMP is also used during status monitoring to assess the health of the node and the objects it contains.



NNM tries community strings serially, in the order listed for the matched region, and uses the first one that works. NNMi tries all configured community strings in parallel and uses the first one that works. Use the best community string where there might be multiple working values.

### Gather from NNM

#### Upgrade tool approach

The `nnmmigration.ovpl` tool collected the community strings from the NNM management station into the `snmpCapture.out` file.

#### Manual approach

The NNM management station has the complete configuration information for SNMP access to the equipment in your environment.

- 1 Export the NNM SNMP configuration by doing one of the following:
  - Open a user interface, select **Options > SNMP Configuration**, and then click **Export**. Name the target file `snmpout.txt`.
  - Run the command:

```
xnmsnmpconf -export > snmpout.txt
```

## NNM SNMP information example

Your output will look something like the following example:

```
10.2.126.75:public:*:::
mytest57.example.net:public:*:::
127.0.0.1:public:*:::
10.97.233.209:mycommstr:*:::
mpls2950.example.net:mycommstr:*:::
mplsce04.example.net:mycommstr:*:::
*.*.*.*:mycommstr*:8:2:900::
```

The target file contains the following fields separated by colons:

```
target:community:proxy(* indicates do not proxy):timeout (tenths
of a second):retries:poll interval (seconds):port:set-community:
```

To see a clear interpretation of the values (but not for use in importing), use the command:

```
xnmsnmpconf -export -verbose
```

For a description of the `ovsnmp.conf` file format, see the `ovsnmp.conf` reference page, or the UNIX manpage, on the NNM management station.

2 Review any configured alternative community strings in the following file:

- *Windows:* %OV\_CONF%\netmon.cmstr
- *UNIX:* \$OV\_CONF/netmon.cmstr

## Replicate to NNMi

### Upgrade tool approach

1 Change to the following directory:

- *Windows:* %NnmDataDir%\tmp\migration\*<hostname>*\SNMP\
- *UNIX:* \$NnmDataDir/tmp/migration/*<hostname>*/SNMP/

2 Create a text file of the NNM community strings:

- *Windows:*

```
%NnmInstallDir%\migration\bin\snmpCapture.ovpl \
snmpCapture.out > snmpout.txt
```
- *UNIX:*

```
$NnmInstallDir/migration/bin/snmpCapture.ovpl \
snmpCapture.out > snmpout.txt
```

3 Follow one of the manual approaches for loading the community strings into NNMi.

4 Configure timeout, retries, and port in the NNMi console.

### Manual approaches

Choose an approach to entering community strings into NNMi. Each of these approaches starts with the list of unique community string values in the `snmpout.txt` file that you created in [step 2](#) on page 22 (for the upgrade tool approach) or [step 1](#) on page 21 (for the manual approach).



The SNMP proxy system and Set community name configuration areas are not transferable.

## Simple manual approach

The easiest approach is to enter all NNM community strings and let NNMi determine the SNMP community string to use for each device. Community string discovery is enabled by default; you can use this feature to expedite the upgrade process.

- 1 Notify your network operations center (NOC) to expect authentication errors during the initial NNMi discovery. NOC personnel can safely ignore these authentication errors during that time.
- 2 Complete one of the following actions:
  - Modify the `snmpout.txt` file to match the format used by NNMi. Then use NNMi to load these values.
  - Use the `snmpout.txt` file as a sample and hand-build the input file for NNMi. Then use NNMi to load these values.
  - Enter the values in the NNMi console by following these steps:
    - a Determine the list of unique community string values in the `snmpout.txt` file.



If you used the upgrade tool approach to create the `snmpout.txt` file from the `snmpCapture.out` file, each community string in the `snmpout.txt` file is unique; you do not need to perform this step.

- *Windows:* Open the `snmpout.txt` file in Microsoft Office Excel. Select the data rows, and then sort on column B.

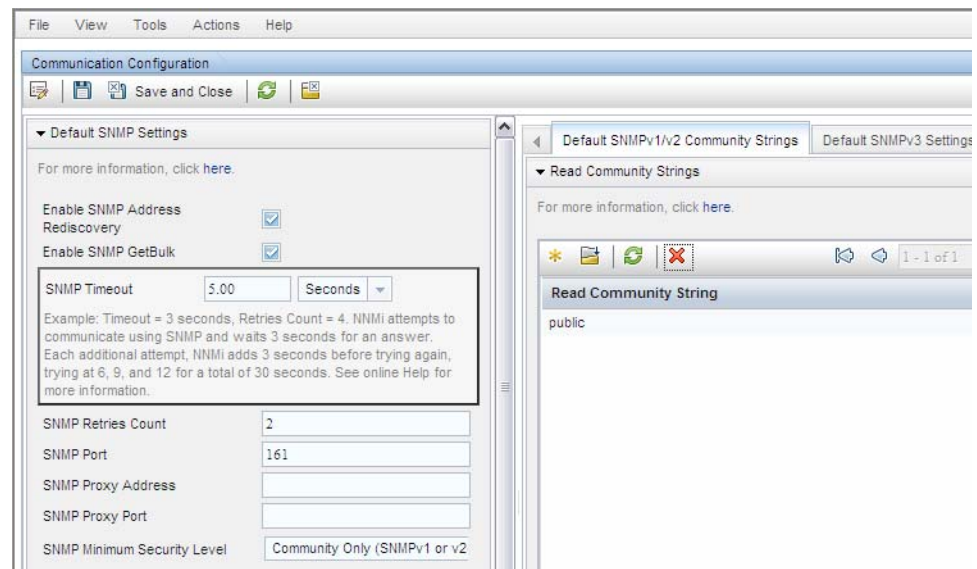
For this example, consider two unique community strings:

```
public
mycommstr
```

- *UNIX:* Run the following command:

```
cut -f 2 -d ':' < snmpout.txt | sort -u
```

- b In the NNMi console, select **Communication Configuration** from the **Configuration** workspace. Enter the unique values on the **Default SNMP v1/v2 Community Strings** tab.
- c Configure timeout, retries, and port.



### Modified simple manual approach

Group community strings by IP region where they are used. Load regional values into the NNMi console, and then let NNMi determine the SNMP community string to use for each device, but with fewer authentication failures than in the simple approach.

- 1 In the `snmpout.txt` file, determine the list of unique values *per IP region* that NNMi is using.
- 2 In the NNMi console, select **Communication Configuration** from the **Configuration** workspace. Create IP Regions, and then enter the community strings for each region.
- 3 Configure timeout, retries, and port.

### Automated manual approach

Convert the `snmpout.txt` file into the format needed by the `nnmcommload.ovpl` command, and then load the specific community string in use for each device.

- 1 Adapt the `snmpout.txt` file for use with the NNMi tool by using one of the following methods:
  - Use an editor to create the file appropriate for NNMi. The result should look similar to:

```
10.2.126.75,public
mytest57.example.net,public
127.0.0.1,public
10.97.233.209,mycommstr
mpls2950.example.net,mycommstr
mplsce04.example.net,mycommstr
```

- *UNIX only:* Run the following command:

```
awk 'BEGIN {FS = ":" }; {printf"%s,%s\n",$1,$2 }' \
<snmpout.txt> mysntp.txt
```

This command works for individual nodes in the file. Trim ranges or wildcards out by hand.

- 2 Run the following command:
 

```
nnmcommload.ovpl -u username -p password -file mysntp.txt
```
- 3 Configure default community strings and community strings for IP ranges in the NNMi console.
- 4 Configure timeout, retries, and port in the NNMi console.

### NNMi console approach

In the NNMi console, select **Communication Configuration** from the **Configuration** workspace. Duplicate the configured values from the `snmpout.txt` file.



## Enhance in NNMi

Enhance your communication access configuration in NNMi with the following information:

- Hostname wildcards (if they suit your environment better than IP ranges)
- ICMP timeout and retries by global default, IP range, and specific node
- Enable or disable SNMP or ICMP access to specific areas of the network
- Tune the options that NNMi uses for selecting a node's management address.
- The preferred management address for specific nodes

## Limit Name Resolution

If you know of limitations in your DNS (or other name resolution) service, you can instruct NNM and NNMi to avoid lookups for those devices. If this task does not apply to your installation, continue to [Customize Device Profiles](#) on page 26.



File name capitalization differs between NNM and NNMi. NNM uses the file name `ipNoLookup.conf`, while NNMi uses the file name `ipnollookup.conf`. NNMi does not correctly interpret anything other than all lowercase characters for this file name.

## Gather from NNM

### Upgrade tool approach

The `nnmmigration.ovpl` tool collected the information about which IP addresses and hostnames to use without DNS lookup from the NNM management station and created one or both of the `ipnollookup.conf` and `hostnollookup.conf` files for configuring NNMi.

### Manual approach

- 1 Review the following file to determine the **addresses** that NNM excludes from address-to-hostname resolution:

- *Windows*: %OV\_CONF%\ipNoLookup.conf
- *UNIX*: \$OV\_CONF/ipNoLookup.conf



If the `ipNoLookup.conf` file does not exist on the NNM management station, there is no configuration to replicate.

- 2 Run the following command to determine the **hostnames** that NNM excludes from name-to-address resolution:

```
snmpnollookupconf -dumpCache > snmpnollookup.out
```



If the `snmpnollookup.out` file is empty, there is no configuration to replicate.

## Replicate to NNMi

## Upgrade tool approach

- 1 If available, edit the `ipnolookup.conf` and `hostnolookup.conf` files created by the `nmmigration.ovpl` tool to delete any references to the NNMi management server:
  - *Windows:*
    - `%NnmDataDir%\tmp\migration\\CONFIG\ipnolookup.conf`
    - `%NnmDataDir%\tmp\migration\\DNS\hostnolookup.conf`
  - *UNIX:*
    - `$NnmDataDir/tmp/migration/<hostname>/CONFIG/ipnolookup.conf`
    - `$NnmDataDir/tmp/migration/<hostname>/DNS/hostnolookup.conf`
- 2 Place the edited configuration files into the following directory:
  - *Windows:* `%NnmDataDir%\conf\`
  - *UNIX:* `$NnmDataDir/shared/nnm/conf/`

## Manual approach

- 1 Add the addresses from the NNM `ipNoLookup.conf` to the following file:
  - *Windows:* `%NnmDataDir%\conf\ipnolookup.conf`
  - *UNIX:* `$NnmDataDir/shared/nnm/conf/ipnolookup.conf`



Do not add the IP address of the NNMi management server.

- 2 Add the hostnames that NNM excludes (from the `snmpnolookup.out` file that you created in [step 2](#) on page 24) to the following file:
  - *Windows:* `%NnmDataDir%\conf\hostnolookup.conf`
  - *UNIX:* `$NnmDataDir/shared/nnm/conf/hostnolookup.conf`



Do not add the hostname of the NNMi management server.

For information about the format of these configuration files, see the `ipnolookup.conf` and `hostnolookup.conf` reference pages, or the UNIX manpages.

## Enhance in NNMi

NNMi does lookups during discovery only. By replicating the NNM no-lookup configuration to NNMi, the spiral discovery operation is automatically enhanced.

In NNMi, you can choose to use the DNS hostname, IP Address, or MIB II `sysName` as the displayed name label. To do so, follow these steps:

- 1 In the NNMi console, select **Discovery Configuration** from the **Configuration** workspace.
- 2 Set your node name preferences in the **Node Name Resolution** area.

## Customize Device Profiles

NNM collects some configuration information directly from SNMP queries to the device. Other information is *derived* from the device's **system object ID** (`sysObjectID`). NNMi maps attributes to a device according to its **device profile**, which is based on the `sysObjectID`. Device profiles group nodes for monitoring, filtering views, and categorizing nodes for discovery maintenance.

The following configuration areas are not transferable:

- Custom symbols
- Custom database fields and default values

### Gather from NNM

- 1 Determine any customizations to the OID files for your version of NNM.
  - NNM 6.4 and earlier used the files `oid_to_sym`, `oid_to_type`, and `HPoid2type` to map a system's `sysObjectID` to database attributes and displayed symbol.
  - NNM 7.x replaces the `oid_to_sym` file with the `oid_to_sym_reg` directory structure.



The `nnmmigration.ovpl` tool copies these files to the `CONFIG` folder within the migration file structure.

### Replicate to NNMi

Because NNMi ships with a large number of device profiles that are preconfigured for known system object IDs, the device profiles that you need might already be available. The **simplest approach** is to start the discovery process, review the results, and then make modifications only as necessary.

#### Best practice

HP recommends that you specify a unique author for each device profile that you create or modify in case you must identify these profiles at a later time.

- 2 In the NNMi console, select **Device Profiles** from the **Configuration** workspace. Locate the entry by **SNMP Object ID** (`sysObjectID`) for each of your customized values.
- 3 Update the device profile configuration as necessary.
  - For the entries that NNMi has available, verify that the configured values match the NNM attributes.
  - For entries that are not included in NNMi, create a new device profile for the `sysObjectID`. Submit an enhancement request to notify HP to add the ID for future releases.

#### Best practice

- 4 After initial discovery, sort the node inventory by device profile to locate the **No Device Profile** nodes.

The **No Device Profile** profile type indicates `sysObjectIDs` that were not previously configured in NNMi. NNMi uses the default monitoring settings for nodes with **No Device Profile**, and these nodes are more difficult to filter.

You can build new device profiles to ensure that configured device profiles exist for all `sysObjectIDs` in the NNMi database.

---

## Phase 3: Upgrade Discovery

Configure the discovery schedule and configuration. NNMi spiral discovery begins immediately after you save one or more discovery seeds.



Configure NNMi to use the appropriate community strings for your network environment before initiating discovery.

After initial discovery, replicate any connections between devices that were configured manually in NNM.

## Schedule Discovery

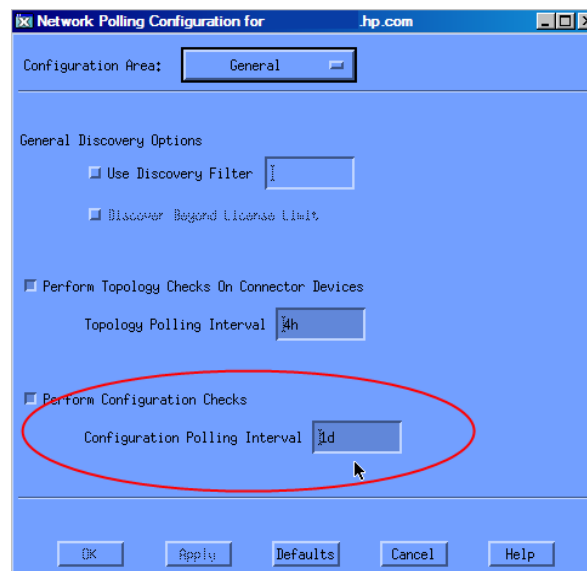
The NNM discovery processes can run independently. To upgrade discovery to NNMi, you only transfer the **interval** at which NNM discovers nodes.

The following schedule configuration areas are no longer used in NNMi and are not transferable:

- Topology checks on connector devices. A topology check now happens automatically whenever NNMi sees a trigger that indicates a possible change.
- Configuration check. A configuration check now happens at the time of a scheduled discovery or with any trigger in NNMi.
- Layer 2 (Extended Topology) discovery behavior. NNMi performs Layer 2 discovery for each device as it is found, so there is no need to schedule this behavior separately.
- Auto-adjusting discovery polling interval.

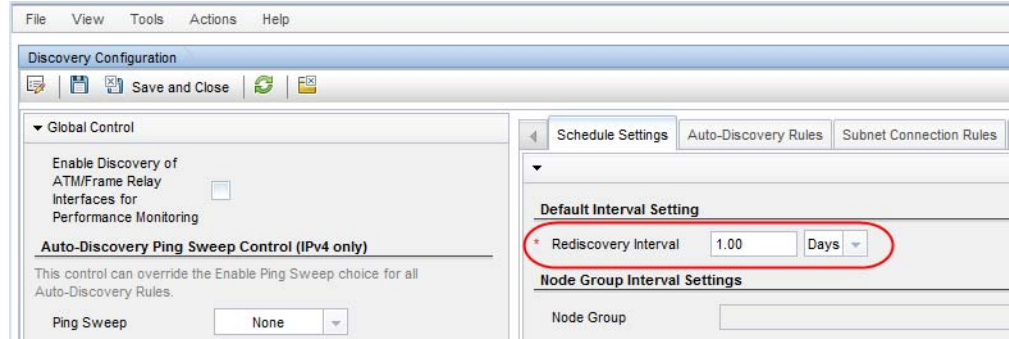
### Gather from NNM

- 1 Determine when NNM performs rediscovery.
  - a In a user interface, select **Options > Network Polling Configuration**.
  - b On the **IP Polling** page, review the **Discovery polling interval** box.
    - If NNM uses a fixed interval, note that value for transfer to NNMi.
    - If NNM uses auto-adjusting intervals, NNM waits a maximum of 24 hours. You can choose to stay with 24 hours, or you can select a new value.
    - If auto-discovery has not been not enabled, determine the interval for **Perform configuration checks** on the **General** page and note that value for transfer to NNMi.



Replicate to NNMi

- In the NNMi console, select **Discovery Configuration** from the **Configuration** workspace, and then set the **Rediscovery Interval** to the value determined in [step 1](#).



Enhance in NNMi

All other configuration updates are automatic and incremental, so configuration is simpler and discovery is more efficient than in NNM.

## Select Your Discovery Method

Determine which model to use for NNMi discovery:

- Seeded discovery with no auto-discovery rules. This type of discovery is bounded by the administrator, who controls what is discovered by adding seeds as necessary. Complete *only* the following task:
  - [Add Seeds to NNMi for Seeded Discovery](#) on page 35
- Automatic discovery based on seeds and auto-discovery rules. Complete both of the following tasks:
  - [Configure Auto-Discovery Rules](#) on page 30
  - [Add Seeds to NNMi for Seeded Discovery](#) on page 35

For more information about the differences between the NNMi discovery methods, see *Determine Your Approach to Discovery* in the NNMi help.



NNM licenses are based on the number of nodes under management (status monitoring). NNMi licenses are based on the number of nodes discovered and placed in the topology (monitored and unmonitored nodes).

While this difference might encourage you to discover fewer nodes, there are advantages to including unmonitored nodes in your database. For example:

- You might want to see a service provider’s access router and your connectivity to it, even if you are not responsible for managing the device.
- Status monitoring algorithms are based on connectivity as seen in the database. Interfaces having no device on the other end of the link *in the database* are unmonitored by default. You might choose to override the default in status monitoring configuration, or you might choose to discover the device. Your choice depends on the balance of interests in your environment. For more information, see “Interfaces to Unmonitored Nodes” in the *NNMi Deployment Reference*.

## Configure Auto-Discovery Rules

NNMi discovery configuration provides an excellent opportunity to consider what you want to manage with NNMi. Before you invest in converting your NNM discovery configuration and filters, consider looking at your current network environment and describing what you want to include in the NNMi topology.

If you do want to invest in direct conversion, NNMi discovery rules encompass two task sets from NNM: extending the scope of discovery and limiting the objects discovered within that scope.



For NNMi configuration, it is important to define all of the rules to extend discovery, limit discovery, or both before entering the seeds, which initiates the discovery process.

The following schedule configuration areas are no longer used in NNMi and are not transferable:

- IPX discovery from Windows
- Discover beyond license limit
- Disable discovery of Layer 2 objects (always enabled for NNMi)
- Discovery exclusions by filtering on attributes other than IP address and sysObjectID (and its derivatives)
- Limiting Layer 2 discovery through `bridge.noDiscover`
- Limiting Layer 2 discovery based on CDP protocol area (such as aggregated ports and vlans)
- Extended Topology zone configuration, which is no longer relevant to NNMi spiral discovery

## Configure Spiral Discovery

NNMi provides two methods for configuring spiral discovery in NNMi: manually loading nodes (for example, from a host file) and using auto-discovery rules.

### Load nodes manually

#### Gather from NNM

- 1 In NNM, find the file that contains the output of the `loadhosts` command. This file lists an IP address and a hostname for each node, plus a subnet mask if one was specified.

NNM loadhosts  
example

An example file for the `loadhosts` command looks similar to the following:

```
10.2.32.201 lnt04.example.net # comment
10.2.32.202 lnt07.example.net # comment
10.2.32.203 lnt03.example.net # comment
10.2.32.204 lnt02.example.net
10.2.32.205 lnt05.example.net
```

#### Replicate to NNMi

- 2 In NNMi, you can use discovery seeds in the same fashion as the NNM `loadhosts` command. To do so, use the `nnmloadseeds.ovpl` command with the `-f` option and specify a seed file.

**Best practice**

Complete all community string configuration before configuring any seeds into NNMi.



If you want the discovery output to be equivalent to NNM loadhosts, disable any auto-discovery rules that are configured in NNMi. To disable an auto-discovery rule, do one of the following:

- Delete the rule from the **Discovery Configuration** form.
- On the **Auto-Discovery Rule** form, clear the **Discover Included Nodes** check box.

The format for the seed file in NNMi is either an IP address or a node name (plus an optional comment) per line. For more information, see the *nnmloadseeds.ovpl* reference page, or the UNIX manpage.

NNMi seed file example

The following example shows an NNMi seed file with the same function as the NNM loadhosts command and a hostfile:

```
10.2.32.201 # comment
10.2.32.202 # comment
1nt03.example.net # comment
1nt02.example.net
10.2.32.205
```

**Best practice**

The following file contains a list of devices from Extended Topology:

- *Windows:* %NnmDataDir%\tmp\migration\*<hostname>*\NNMET\hosts.nnm
- *UNIX:* \$NnmDataDir/tmp/migration/*<hostname>*/NNMET/hosts.nnm

You can copy the first field (IP address) or second field (nodename) to create a seedfile for NNMi.

On UNIX, you can run the following command to create a file of the node names:

```
cut -f 2 hosts.nnm
```

**Best practice**

NNMi always favors the loopback address as the management address. If you do not use loopback addresses, NNMi probably (but not always) uses the seed address as the management address. Therefore, it is a good practice to populate the hostfile with preferred IP addresses. If you use hostnames, verify that the DNS resolves to the preferred management address, which still does not guarantee that NNMi will use this address as the management address. For more information about management address selection, see *Discovery Node Name Choices* in the NNMi help.

Use auto-discovery rules

**Gather from NNM**

- 1 Determine whether a discovery filter was used for NNM. In NNM, one discovery filter applied to the entire scope of discovery.
  - a Open an NNM user interface.
  - b Select **Options > Network Polling Configuration**.
  - c On the **General** page, review the **Use filter** check box and, if selected, note the discovery filter in use. If no filter is in use, continue with [Add Seeds to NNMi for Seeded Discovery](#) on page 35.
  - d Locate the discovery filter in the following file:
    - *Windows:* %OV\_CONF%\C\filters
    - *UNIX:* \$OV\_CONF/C/filters



- e Review the discovery filter logic carefully.

For NNMi, you can filter on IP address ranges and system object ID ranges. You might be able to translate some attributes, such as hostname wildcards to IP ranges or vendor names to system object ID ranges.

NNM discovery filter example

The following example shows an NNM filter, including Routers, Bridges, Nokia\_Firewalls, NetBotz, and NetsNSegs. You can see that NetBotz and Nokia firewalls are defined through their sysObjectID.

```
Nokia_Firewalls "Nokia Firewalls"
{ ( isNode && ( "SNMP sysObjectID" ~ .1.3.6.1.4.1.94.1.21.2.1.1 ) )
  ||
  ( isNode && ( "SNMP sysObjectID" ~ .1.3.6.1.4.1.94.1.21.2.1.9 ) )
  ||
  ( isNode && ( "SNMP sysObjectID" ~ .1.3.6.1.4.1.94.1.21.2.1.10 ) )
  ||
  ( isNode && ( "SNMP sysObjectID" ~ .1.3.6.1.4.1.94.1.21.2.1.11 ) )
  ||
  ( isNode && ( "SNMP sysObjectID" ~ .1.3.6.1.4.1.94.1.21.2.1.12 ) )
  ||
  ( isNode && ( "SNMP sysObjectID" ~ .1.3.6.1.4.1.94.1.21.2.1.138 ) )
}
```

```
NetBotz "NetBotz"
{ isNode && ( "SNMP sysObjectID" ~ .1.3.6.1.4.1.5528.* ) }
```

```
My_NetInfrastructure "My Network Infrastructure"
{ Routers || Bridges || Nokia_Firewalls || NetBotz || NetsNSegs }
```

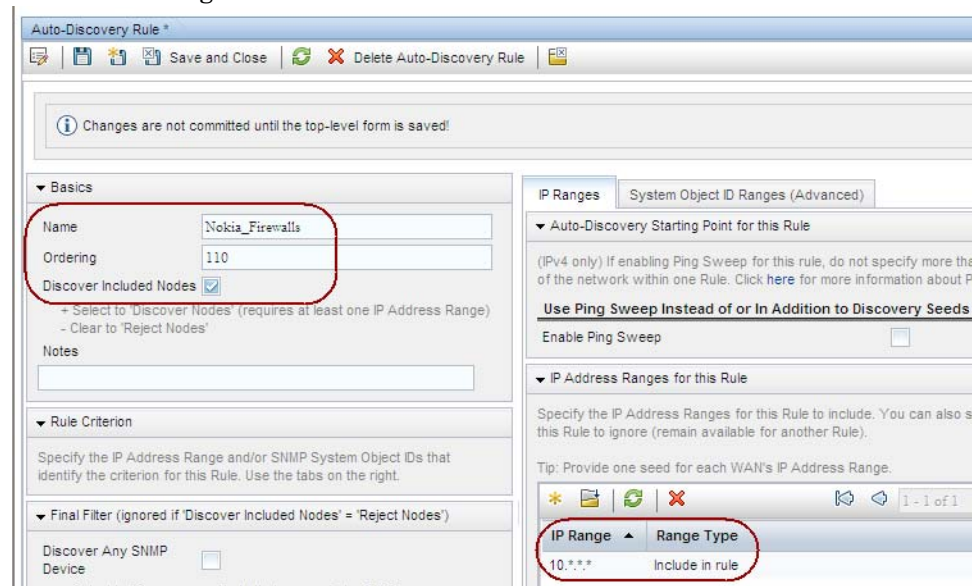
Replicate in NNMi

- 2 Enter the discovery filters in the NNMi console.

NNMi discovery filter entry example

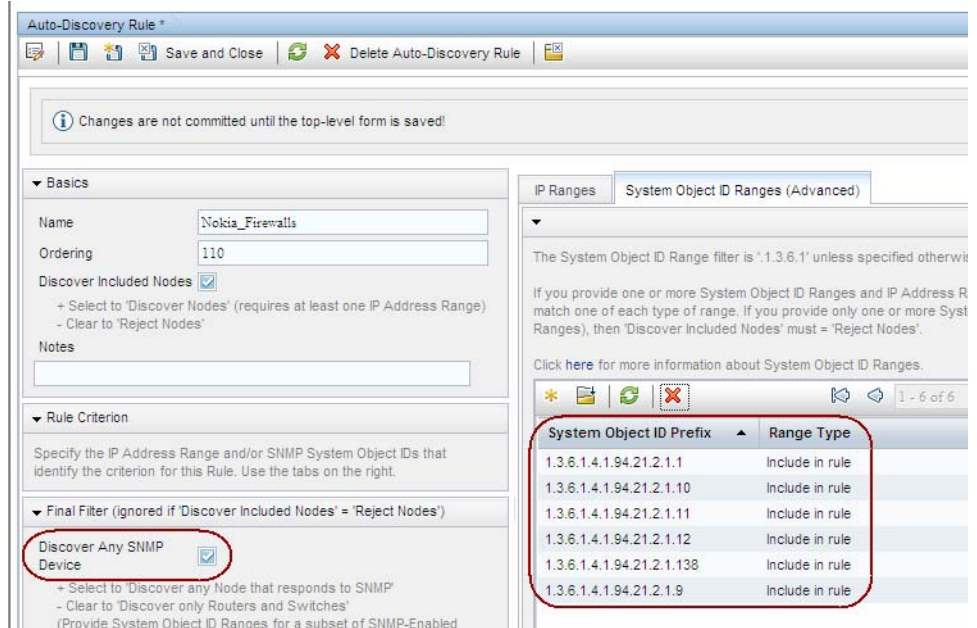
For example, to transfer the NNM filter shown in the [NNM discovery filter example](#) on page 32 to NNMi, you would define three auto-discovery rules: one rule for Nokia firewalls, one rule for NetBotz devices, and a final rule for Routers and Switches (same as Bridge in NNM 7.x). NNMi does not require NetsNSegs. For this example, assume that the range of the network to be discovered is 10.\*.\*.\*.

- a For Nokia firewalls, enter a rule name (Nokia\_Firewalls), and then enter the network IP range 10.\*.\*.\*.

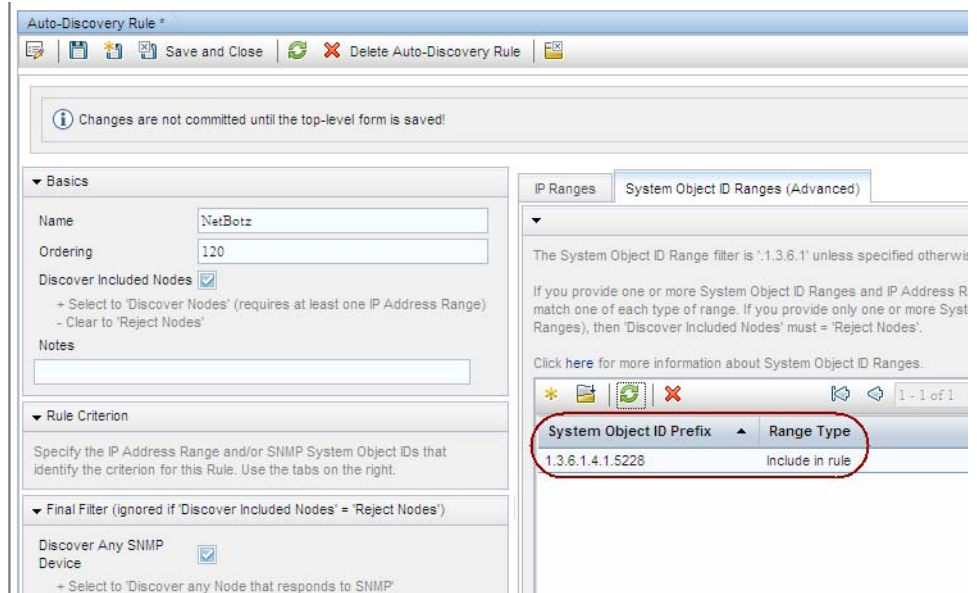




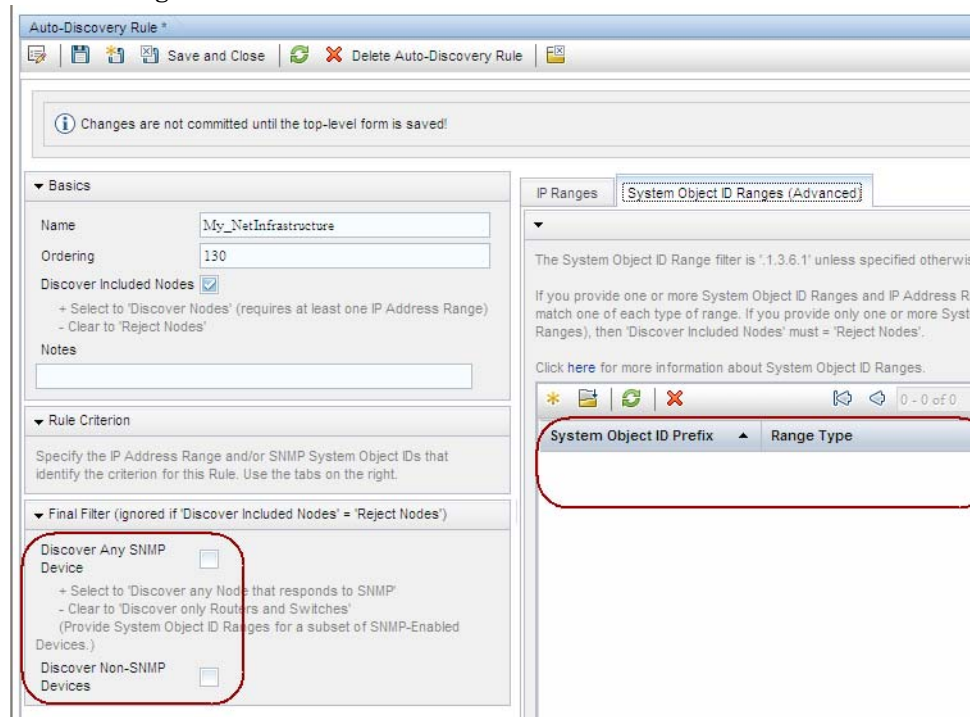
- b Enter each sysObjectID (do not enter the leading period), and then select the **Discover Any SNMP Device** check box. (By default, NNMi only discovers switches and routers. Because these devices might not be marked as switches or routers, select the **Discover Any SNMP Device** check box when specifying sysObjectIDs.)



- c Enter the NetBotz rule. This rule uses a wildcard in NNM: .1.3.6.1.4.1.5528.\*. In NNMi, the asterisk (\*) is implied and not required.



- d The final rule is for switches and routers. Because NNMi discovers these devices by default, do not specify system object IDs. Only specify the IP address range.



## Exclude Addresses from Discovery

You can specify IP addresses that are never discovered. Do not populate the Excluded IP Addresses filter with the addresses associated with SNMPv1/SNMPv2c agents or SNMPv3 engines (the management addresses).



If the `netmon.noDiscover` file does not exist on the NNM management station, there is no configuration to replicate. You can follow the NNMi console approach to specify IP addresses that NNMi should not discover.

### Gather from NNM

#### Upgrade tool approach

The `nnmmigration.ovpl` tool collected the `netmon.noDiscover` file from the NNM management station.

#### Manual approach

Review the following file to determine the IP addresses that NNM excludes from discovery:

- **Windows:** `%OV_CONF%\netmon.noDiscover`
- **UNIX:** `$OV_CONF/netmon.noDiscover`

### Replicate to NNMi

#### Upgrade tool approach

1 Change to the following directory:

- **Windows:** `%NnmDataDir%\tmp\migration\\CONFIG\conf\`
- **UNIX:** `$NnmDataDir/tmp/migration/<hostname>/CONFIG/conf`

2 Import the IP addresses in the `netmon.noDiscover` file into the NNMi database:

- *Windows:*

```
%NnmInstallDir%\bin\nnmdiscocfg.ovpl -excludeIpAddrs \  
-f netmon.noDiscover
```

- *UNIX:*

```
$NnmInstallDir/bin/nnmdiscocfg.ovpl -excludeIpAddrs \  
-f netmon.noDiscover
```

#### NNMi console approach

In the NNMi console, select **Discovery Configuration** from the **Configuration** workspace. On the **Excluded IP Addresses** tab, enter the IP addresses from the `netmon.noDiscover` file.

## Add Seeds to NNMi for Seeded Discovery

### Gather from NNM

#### Upgrade tool approach

The `nnmmigration.ovpl` tool collected the list of devices in the NNM database from the NNM management station into the `topology.out` file.

#### Manual approach

Determine the exact list of devices in the NNM database by running the following command:

```
ovtopodump > topology.out
```

### Replicate in NNMi

1 Locate the `topology.out` (export) file from NNM.

- For the upgrade tool approach, this file is located as follows:

- *Windows:*

```
%NnmDataDir%\tmp\migration\<hostname>\TOPO\topology.out
```

- *UNIX:*

```
$NnmDataDir/tmp/migration/<hostname>/TOPO/topology.out
```

- For the manual approach, this file is in the local directory.

2 Copy and edit the `topology.out` file from NNM, or retype the entries into a file for importing into NNMi. The new file should have one explicit IP address or hostname per line. You do not need to specify a subnet prefix because NNMi determines the subnet automatically.

NNMi seed file  
example

```
10.2.32.201 # comment  
10.2.32.202 # comment  
lnt03.example.net # comment  
lnt02.example.net  
10.2.32.205
```



Alternatively, you can add this list of nodes by using the NNMi console.

- 3 Run the following command:

```
nnmloadseeds.ovpl -f newSeedfile
```

For more information, see the *nnmloadseeds.ovpl* reference page, or the UNIX manpage.

NNMi begins to discover the devices associated with these seeds immediately and implements the existing device profiles (and node groups, such as node groups for status monitoring). NNMi spiral discovery is ongoing. For information about how to determine discovery status, see “Check Discovery Progress” in the *NNMi Installation Guide*.

## Customize Connectivity

In certain circumstances where device information is limited, NNM’s Extended Topology might not accurately discover and model every connection in a network. As a result, you might see no connections where you know connections exist or connections indicated where you know none exist. The remedy for this situation is to create the correct connections manually. You can replicate the connection configuration in NNMi.

### Gather from NNM

- 1 Review the following file to determine whether manual connections have been configured in NNM:
  - *Windows*: %OV\_CONF%\nnmet\connectionEdits
  - *UNIX*: \$OV\_CONF/nnmet/connectionEdits

The use of these files is documented in the *Using Extended Topology* manual or the white papers directory.

### NNM connection example

The following example shows how to create two connections in NNM 7.x. One connection is based on `ifAlias`, and the other is based on `ifIndex` (along with board).

```
N1.example.net[ifAlias:MyAlias],N2.example.net[ifAlias:MyOtherAlias]
Y1.example.net[ 0 [ 999 ]],Y2.example.net[ 0 [ 2 ]]
```

### Replicate to NNMi

- 2 Use the `nnmconnedit.ovpl` tool to make connection edits in NNMi. The file format is completely different from that used by NNM.
  - a Generate a connection template file by running the following command:

```
nnmconnedit.ovpl -t add
```

For more information, see the *nnmconnedit.ovpl* reference page, or the UNIX manpage.

- b Edit the template file (`add.xml`) to change or add connections. Use the documentation in the file for the syntax of the new file.

### NNMi connection example

The following example shows the NNMi equivalent to the [NNM connection example](#) on page 36:

```
<connectionedits>
  <connection>
    <operation>add</operation>
    <node>N1.example.net</node>
    <interface>MyAlias</interface>
    <node>N2.example.net</node>
    <interface>MyOtherAlias</interface>
```

```

        </connection>
        <connection>
            <operation>add</operation>
            <node>Y1.example.net</node>
            <interface>999</interface>
            <node>Y2.example.net</node>
            <interface>2</interface>
        </connection>
    </connectionedits>

```

- c Load the new connection information into the database by running the following command:

```
nnmconnedit.ovpl -f add.xml
```

- d In the NNMi console, select **Layer 2 Connections** from the **Inventory** workspace to verify the results.

---

## Phase 4: Upgrade Status Monitoring

In NNM 6.x, the `netmon` process performs status monitoring. In NNM 7.x, the `netmon` process or APA performs status monitoring.

- The `netmon` process models devices, such as nodes that contain interfaces, and applies polling parameters primarily at the node level.
- APA models addresses, interfaces, aggregated interfaces, boards, and nodes. APA can apply polling parameters at any of these levels.

With NNMi, you can apply polling parameters at the node, interface, and default levels.

NNMi does not provide special handling for DHCP nodes, so this configuration is not transferable.

### Set Polling Intervals

#### Gather from NNM [NNM netmon polling process](#)

If the `netmon` process is your NNM general poller, obtain the polling intervals from the NNM user interface.

#### [NNM APA polling process](#)

NNM paConfig.xml  
example

If APA is your NNM general poller, find the `paConfig.xml` file and determine the current polling intervals. For example:

```

<classSpecification>
  <filterName>isRouter</filterName>
  <parameterList>
    <parameter>
      <name>interval</name>
      <title>Interval to Poll Device</title>
      <description>
        The interval for which the device will be polled
        in seconds.
      </description>
    </parameter>
  </parameterList>
</classSpecification>

```

```

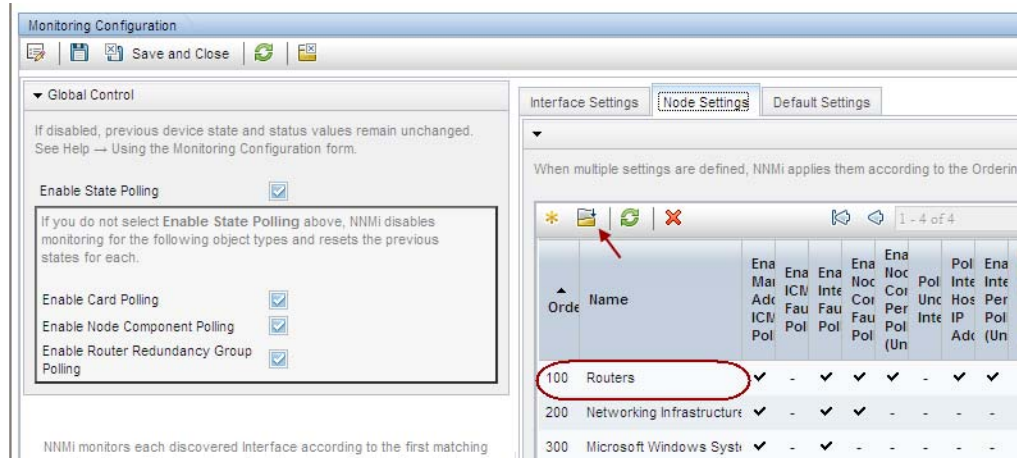
</description>
<varValue>
  <varType>Integer</varType>
  <value>300</value>
</varValue>
</parameter>
.
.
.
</parameterList>
</classSpecification>

```

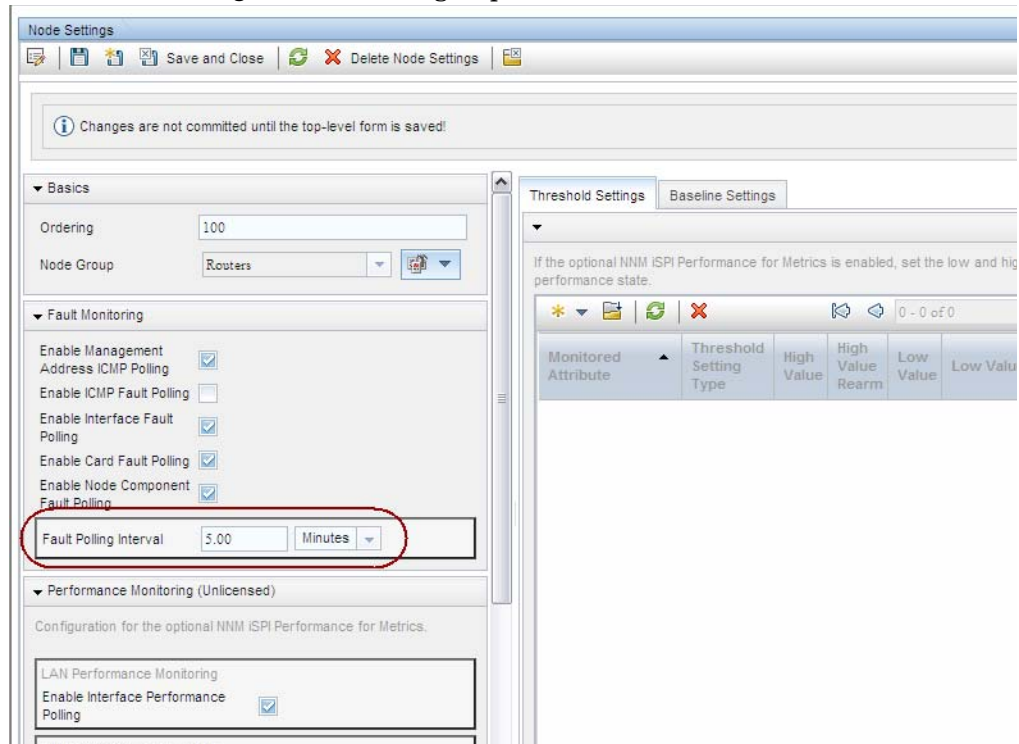
Replicate to NNMi NNMi polling process

NNMi status monitoring configuration is based on groups of nodes, groups of interfaces, or both.

- 1 In the NNMi console, select **Monitoring Configuration** from the **Configuration workspace**.
- 2 On the **Node Settings** tab, open a node group.



- 3 Set the **Fault Polling Interval** for the group.





## Select Polling Protocol

### Gather from NNM

#### NNM netmon polling process

By default, the netmon process uses ICMP to poll each address (equated with an interface). NNM can be configured so that the netmon process uses SNMP rather than ICMP (it never uses both) for some devices. To determine whether some areas are using ICMP, review the following file:

- **Windows:** %OV\_CONF%\netmon.snmpStatus
- **UNIX:** \$OV\_CONF/netmon.snmpStatus

#### NNM APA polling process

APA uses a combination of SNMP and ICMP for polling. In APA, the polling policies are applied to nodes or interfaces, which are grouped by filters. The filters are defined in the TopoFilters.xml file. The polling policies are defined in the paConfig.xml file.

### Replicate in NNMi

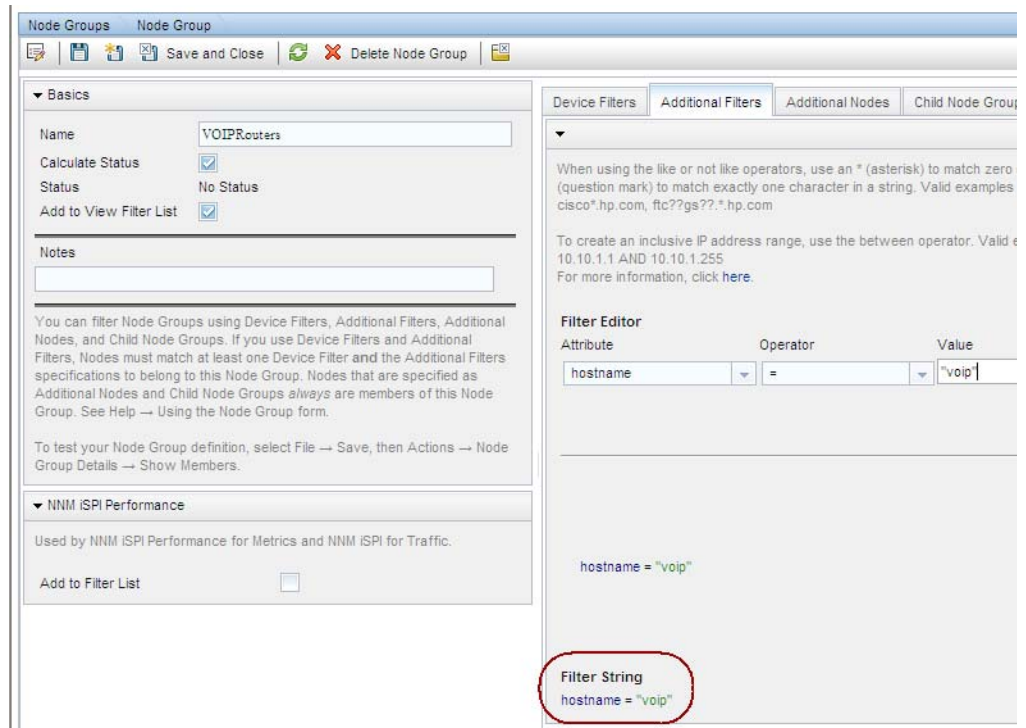
#### NNMi polling process

In NNMi, the nodes and interface collections are defined as node groups and interface groups. Polling policies are applied to node groups and interface groups on the **Monitoring Configuration** form.

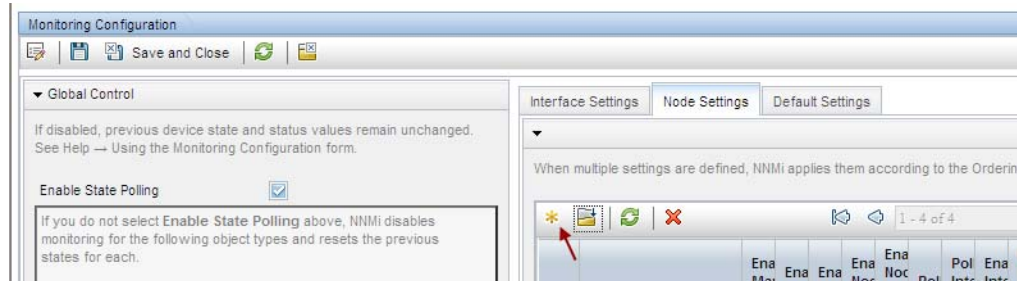
NNMi polling configuration example

For example, to configure polling (using SNMP and ping) for a collection of VOIP routers, follow these steps:

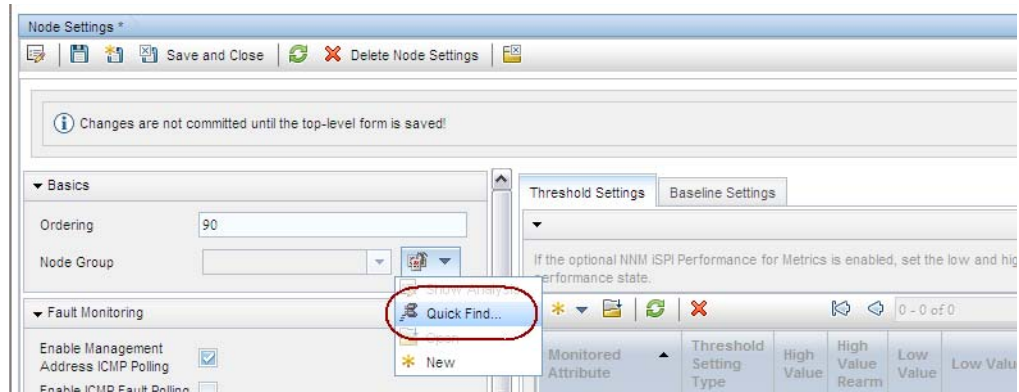
- 1 Using the **Node Group** form, create a node group that identifies the VOIP routers. Save and close this form.



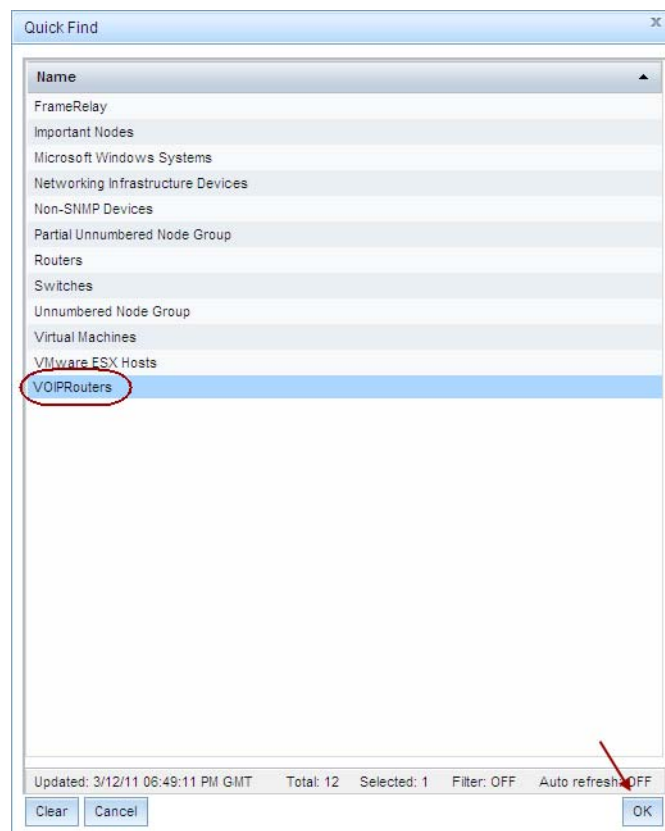
- 2 On the **Monitoring Configuration** form, on the **Node Settings** tab, click **New**, as shown here.



- 3 Specify an ordering value, and then select quick find for the **Node Group** field, as shown here.

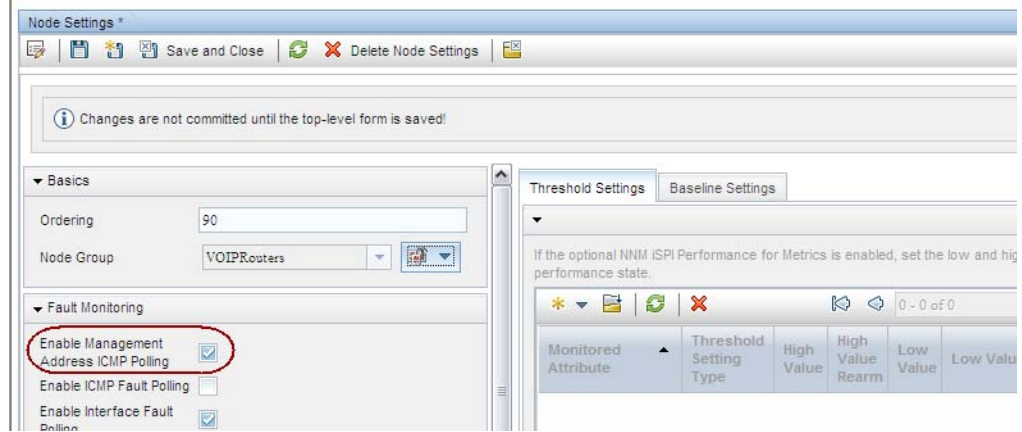


- 4 Select the **VOIPRouters** node group, and then click **OK**, as shown here.





- Verify that the **Enable Management Address ICMP Polling** check box is selected, as shown here. Save and close the form.



## Configure Critical Nodes

By default, NNMi provides a node group for important nodes. This node group functions in the same way as the critical nodes list in NNM.

When important nodes are down or unreachable, NNMi shows node status as critical and generates a NodeDown incident.

### Gather from NNM

#### NNM netmon polling process

If NNM uses netmon for status monitoring, NNM is not configured for critical nodes. You can create a new critical node configuration in NNMi.

#### NNM APA polling process

Review the following file to determine which nodes are designated as critical for APA:

- Windows:** %OV\_CONF%\nnmet\topology\filter\CriticalNodes.xml
- UNIX:** \$OV\_CONF/nnmet/topology/filter/CriticalNodes.xml

### NNM CriticalNodes.xml example

The CriticalNodes.xml file should resemble the following example:

```
<HostIDs xmlns="http://www.hp.com/openview/NetworkTopology/TopologyFilter" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.hp.com/openview/NetworkTopology/TopologyFilter HostIDFile.xsd">
  <DNSName>router1.example.net</DNSName>
  <DNSName>router7.example.net</DNSName>
  <DNSName>MPLSRtr*.example.net</DNSName>
</HostIDs>
```

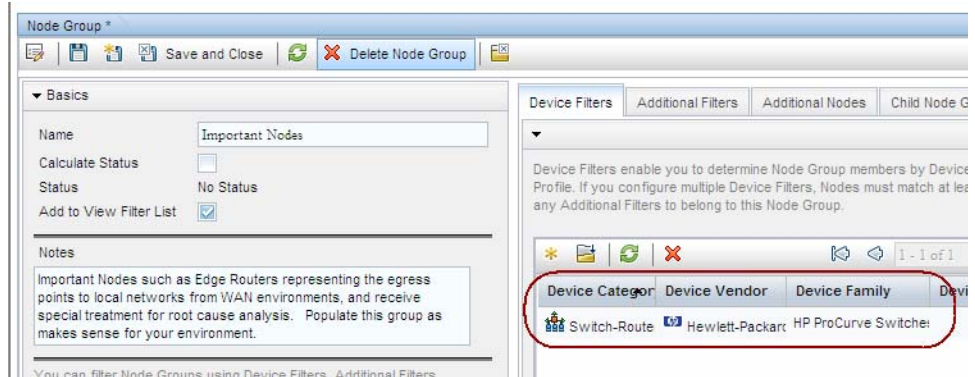
### Replicate to NNMi

#### NNMi polling process

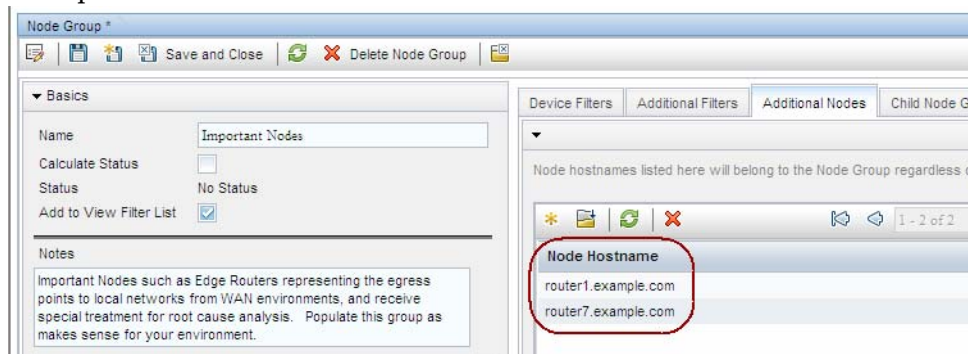
- In the NNMi console, select **Node Groups** from the **Configuration** workspace.
- Open the **Important Nodes** group.

- 3 Add the important nodes to the group by hostname wildcard, device filter, or specific nodes, as shown here.

- a Add a device filter.



- b Add specific nodes. Save and close the form.

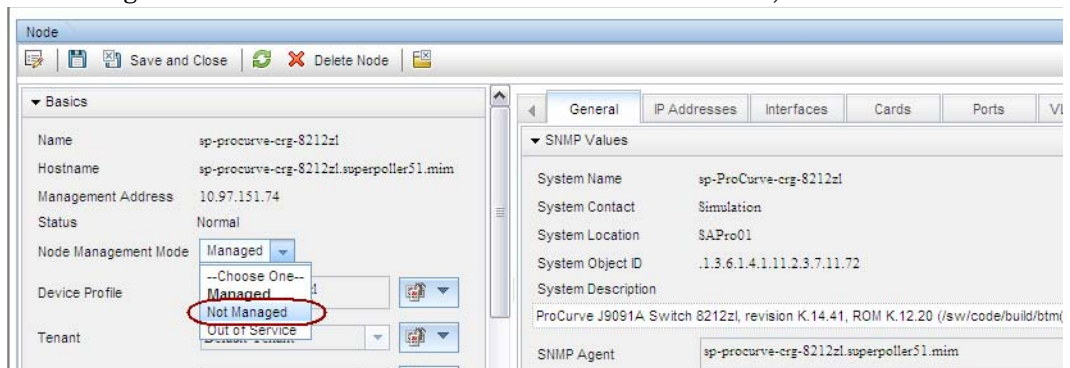


## Exclude Objects from Status Polling

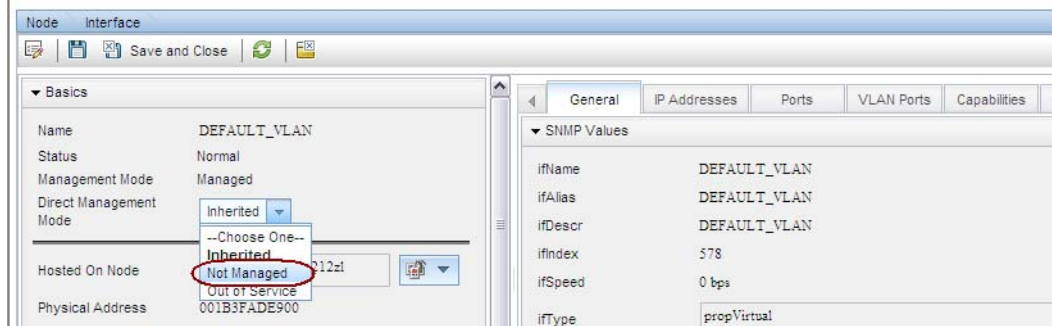
In NNM, most activities that stop nodes or interfaces from being monitored (set them to an UNMANAGED state) are completed through manual intervention in the NNM user interface.

NNMi streamlines the process of unmanaging objects. It is possible that the new product defaults match what you used to do manually (for example, only polling uplinks); however, managing settings through node groups and interface groups makes it easier to update settings automatically.

Occasionally you might need to mark a node or interface as **Not Managed**. You can set the management mode of an individual node on the **Node** form, as shown here:



You can set the management mode of an individual interface on the **Interface** form, as shown here:



## Phase 5: Upgrade Event Configuration and Event Reduction

NNM analyzes all sources of incoming events (traps from managed devices, internal process communication, forwarded events) using an extended SNMPv2c format. Each event has an event object identifier, a name, and configuration parameters.

NNMi handles sources of events differently. Traps from devices and events forwarded from NNM management stations are in the SNMPv2c format. NNMi internal process communications use a new (non-trap) mechanism to significantly improve overall performance. Unrecognized events are now discarded by default. If the NNM management station forwards events to the NNMi management server, ensure that NNMi contains incident definitions for all forwarded events.

Some Composer correlation types (suppress, enhance, transient, multisource) are no longer used in NNMi and are not transferable.

### Display Traps from Devices

You can configure NNMi to display traps from devices in a way that is similar to the NNM environment.

NNMi contains default configurations for many of the common SNMP and vendor traps shipped with NNM. You can update NNMi with any customizations of these traps.

For a list of variables available for messages and automatic actions, see *Configure an Action for an Incident* and *Valid Parameters for Configuring Incident Actions* in the NNMi help.

#### Gather from NNM Upgrade tool approach

The `nnmmigration.ovpl` tool collected the `trapd.conf` file and the MIBs that have been loaded into NNM.

#### Manual approach

Determine whether the NNM configuration includes customized traps. Note any customizations made to category, severity, display message, or automatic actions.

#### Replicate to NNMi Upgrade tool approach

- 1 Change to the following directory:

- *Windows:* %NnmDataDir%\tmp\migration\\CONFIG\conf\
- *UNIX:* \$NnmDataDir/tmp/migration//CONFIG/conf/

## 2 Load the NNM MIBs into NNMi:

- *Windows:*

```
%NnmInstallDir%\migration\bin\nnmibmigration.ovpl \
-file snmpmib -u <user> -p <password>
```

- *UNIX:*

```
$NnmInstallDir/migration/bin/nnmibmigration.ovpl \
-file snmpmib -u <user> -p <password>
```



This step only loads TRAP-TYPE and NOTIFICATION-TYPE MIB entries. NNMi does not use other MIB variables.

## 3 Load the NNM event definitions that are not included with NNMi:

- *Windows:*

```
%NnmInstallDir%\migration\bin\nnmtrapdload.ovpl \
-loadTrapd <lang>\trapd.conf -authorLabel NNM_migration \
-authorKey com.domain.nnmUpgrade -u <user> -p <password>
```

- *UNIX:*

```
$NnmInstallDir/migration/bin/nnmtrapdload.ovpl \
-loadTrapd <lang>/trapd.conf -authorLabel NNM_migration \
-authorKey com.domain.nnmUpgrade -u <user> -p <password>
```

### Best practice

It is recommended that you specify a unique author for this operation in case you must identify these event definitions at a later time. Create an author in the NNMi console, and then use those values for the author label and key in the `nmtrapdload.ovpl` command.

### Manual approach

- 1 Download the vendor MIB files to the NNMi management server.
- 2 Load the MIB files into the NNMi database. For each MIB, run the following command:

```
nnmloadmib.ovpl -loadMib <mibFile>
```

- If one MIB has a dependency on another MIB file, load that MIB file first.
- To see which MIBs are already loaded, use the command:

```
nnmloadmib.ovpl -list
```

For more information, see the `nnmloadmib.ovpl` reference page, or the UNIX manpage.

- 3 Load trap definitions from the MIB files. For each MIB, run the following command:

```
nnmincidentcfg.ovpl -loadTraps <mib_module_name>
```

For more information, see the *nnmincidentcfg.ovpl* reference page, or the UNIX manpage.



This step loads TRAP-TYPE and NOTIFICATION-TYPE MIB entries only. NNMi does not use other MIB variables.

- 4 In the NNMi console, from the **Configuration** workspace, select **SNMP Trap Configurations** (under **Incidents**). The **SNMP Trap Configurations** table displays the incidents configured for received SNMP traps.
- 5 Customize the trap incidents to match those in NNM. You can create categories as needed on the trap configuration form.
- 6 (Optional) In addition to setting default **Severity**, **Category**, and **Message Format**, set a default **Family**.
- 7 (Optional) Classify the trap as a root cause, so that it will appear in the **Open Root Cause Incidents** view.

**Enhance in NNMi**

## Customize Display of NNMi-Generated Management Events

In NNMi, event configuration is simplified because the NNMi causal engine generates a more concise root cause than NNM.

You can modify the incidents generated with NNMi so that they have a similar appearance to NNM alarms. For example, you can customize the NNMi NodeDown incident message to be similar to the message for an NNM NodeDown alarm.

**Gather from NNM**

**Replicate to NNMi**

- 1 In NNM, determine any customizations to the events configuration.
- 2 In the NNMi console, from the **Configuration** workspace, select **Management Event Configurations** (under **Incidents**).
- 3 Locate the new incident configuration by name rather than event number.
- 4 *Optional.* Customize event displays to match those in NNM by creating categories on the trap configuration form.
- 5 In addition to setting default **Severity**, **Category**, and **Message Format**, you can set a default **Family**.

## Block/Ignore/Disable Traps

NNM provides several levels of event processing:

- Block traps as they come into `ovtrapd`
- Process, but do not store or display traps or events labeled `IGNORE`
- Store and process (correlate) events labeled `LOGONLY`, but never display them
- Store, process, and display an event into a category
- Traps that arrive without a configuration appear in the Alarm Browser as `No format in trapd.conf for...` and are stored in the database

NNMi has a simpler approach. A *disabled* event or trap is not stored, processed, or displayed. An *enabled* event or trap is fully stored, processed, and displayed. Any event for which NNMi does not have a configuration is blocked.

## Gather from NNM

### Upgrade tool approach

The `nnmmigration.ovpl` tool collected the `ovtrapd.conf` file.



The `ovtrapd.conf` file is available for NNM 7.51 or higher. The upgrade tool approach does not consider trap definitions. You might want to manually port the LOGONLY configuration for NNM traps.

### Manual approach

- 1 Determine any customizations that ignore traps or set traps to LOGONLY.
- 2 Determine whether NNM uses the trap filtering mechanism (`ovtrapd.conf`, new with NNM 7.51).

## Replicate to NNMi

### Upgrade tool approach

- 1 Change to the following directory:
  - *Windows:* `%NnmDataDir%\tmp\migration\\CONFIG\conf\`
  - *UNIX:* `$NnmDataDir/tmp/migration/<hostname>/CONFIG/conf/`
- 2 Copy the non-commented lines from the NNM `ovtrapd.conf` file into the `nnmtrapd.conf` file by entering the following command:
  - *Windows:*

```
%NnmInstallDir%\migration\bin\nnmtrapdMerge.ovpl \
ovtrapd.conf
```
  - *UNIX:*

```
$NnmInstallDir/migration/bin/nnmtrapdMerge.ovpl \
ovtrapd.conf
```

### Manual approach

- 1 In the NNMi console, from the **Configuration** workspace, select **SNMP Trap Configurations** (under **Incidents**). Locate any events that you do not want to receive or display, and clear the **Enabled** check box for those events.
- 2 To block traps from specific IP addresses, edit the following file to update NNMi with the trap filtering information from NNM:
  - *Windows:* `%NnmDataDir%\shared\nnm\conf\nnmtrapd.conf`
  - *UNIX:* `$NnmDataDir/shared/nnm/conf/nnmtrapd.conf`
- 3 Use the `nnmtrapconfig.ovpl` command to enable trap blocking and to configure the rates and thresholds for trap blocking.
 

For information about using this command, see the `nnmtrapconfig.ovpl` reference page, or the UNIX manpage.

## Configure Lifecycle Transition Actions

NNMi 9.20 does not include up management event incidents. If you need notification that a node is up, associate a lifecycle transition action with the CLOSED lifecycle state of the NodeDown incident.



Integrations that use the NNMi northbound interface (including the NNMi Integration Module for Netcool Software), can receive traps that indicate when a NodeDown incident has been closed.

**Gather from NNM**

**Replicate to NNMi**

- 1 Determine any automatic actions that have been configured for NNM.
- 2 Copy action scripts from the NNM management station to the NNMi management server, where file location is not important.
- 3 In the NNMi console, from the **Configuration** workspace, select **SNMP Trap Configurations** (under **Incidents**).
- 4 For each NNM event with an automatic action, configure the corresponding NNMi incident with that action (on the **Actions** tab).

For most events, to match the behavior of NNM, set the **Lifecycle State** to **Registered**.

For NNM up events, configure the corresponding NNMi down incident. For example, for the NNM NodeUp event, associate the action with the CLOSED lifecycle state of the NNMi NodeDown management event incident.

- 5 For each action script, verify the script functionality:
  - Does the script use parameters to input values from the incident? If so, update these parameters to the NNMi names. For the valid NNMi parameters, see *Valid Parameters for Configuring Incident Actions* in the NNMi help.
  - Does the script call any commands? If so, are these commands available on the NNMi management server, and do they produce the same output as on the NNM management station?

For information about migrating NNM-provided commands to NNMi-provided commands, see [Phase 7: Upgrade Custom Scripts](#) on page 53.

- Does the script logic work correctly on the NNMi management server?

**Enhance in NNMi**

- 6 Note the following NNMi configuration techniques:
  - You can configure more than one automatic action to occur when an event arrives (REGISTERED).
  - You can configure one or more additional actions for each of the other lifecycle states (IN PROGRESS, COMPLETED, CLOSED).
  - You can pass more incident attributes to the command than in NNM.
  - The procedure is simplified because you do not need to register commands in a separate configuration file before NNMi can run them.


## Configure Additional (Manual) Actions

NNM provides operator actions or additional actions that are available from the menu in the Alarms Browser. You might be able to simulate the NNM actions with launch actions that are available from the NNMi console menu.

**Gather from NNM**

**Replicate to NNMi**

- 1 Determine any custom operator actions in NNM.
- 2 For these custom actions, determine how to transfer them to be available as URLs. For a quick-reference list of all URL choices for launching NNMi, see **Help > NNMi Documentation Library > Integrate NNMi Elsewhere with URLs** in the NNMi console.
- 3 In the NNMi console, from the **Configuration** workspace, select **Menu Items** (under **User Interface**).

- 4 On the **Menu Items** table, click  **New**.
- 5 On the **Menu Item** form, enter the **Menu Item Label**, a **Unique Key**, **Ordering**, and **Selection Type**.
- 6 On the **Menu Item Contexts** tab, click **New**.
- 7 On the **Menu Item Context** form, for **Menu Item Action**, select **New Launch Action**.
- 8 On the **Launch Action** form, enter a **Name** and the **Full URL** for the action.
- 9 **Save and Close** back to the NNMi console.

## Event Correlation: Repeating Events

NNM mechanisms use either the first or last event as the parent when deduplicating events.

NNMi creates a new parent with the **Dedup Stream Correlation** correlation nature. The parent incident appears in the **All Incidents** incident view. The original events appear in their configured incident views.

### Gather from NNM

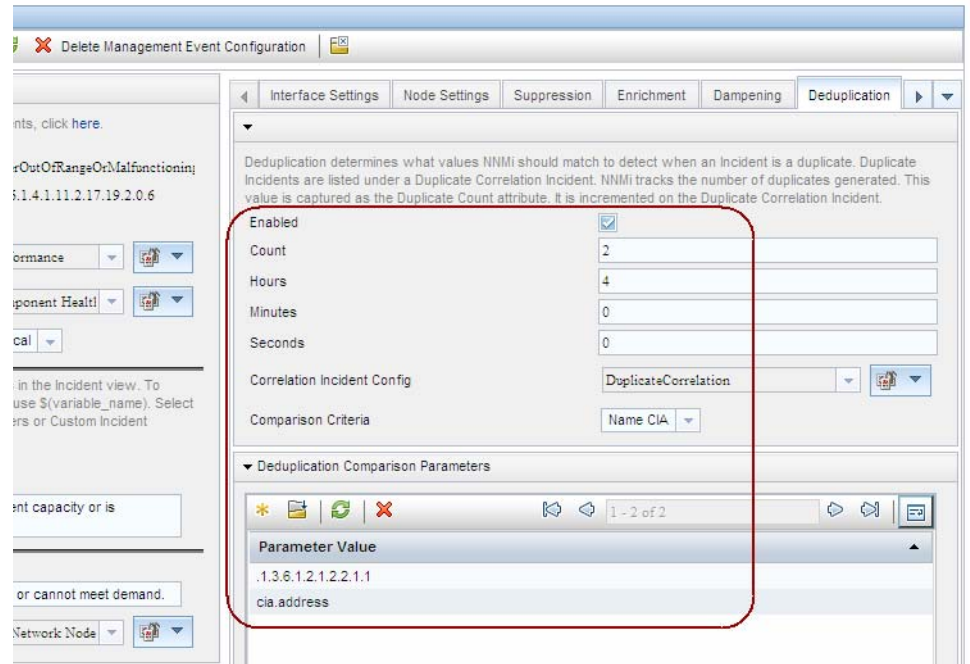
- 1 Determine whether the `RepeatedEvents` correlation is in use for NNM.
- 2 Determine whether the `Repeated` correlator is in use for NNM.
- 3 Determine whether deduplication is in use (`dedup.conf` file).

### Replicate to NNMi

- 4 In the NNMi console, from the **Configuration** workspace, select **SNMP Trap Configurations**, **Management Event Configurations**, or **Remote NNM 6.x/7.x Event Configurations** (under **Incidents**).
- 5 Open the incident type to be deduplicated.
- 6 On the **Deduplication** tab, do the following:
  - a Select **Enabled** to enable monitoring.
  - b Set the count window.
  - c Set the time window (**Hours**, **Minutes**, and **Seconds** fields).
  - d Select a management event incident type (for example, **DuplicateCorrelation**) as the new parent event (**Correlation Incident Config**).
  - e Define the **Comparison Criteria**.



For more information, see *Configure Deduplication for an SNMP Trap Incident* in the NNMi help.



## Event Correlation: Counting the Rate

NNM mechanisms use either the first or last event as the parent when deduplicating events.

NNMi creates a new parent with the **Rate Stream Correlation** correlation nature. The parent incident appears in the **All Incidents** incident view. The original events appear in their configured incident views. NNMi has sustained rate behavior equivalent to the rolling time window in NNM.

**Gather from NNM**  
**Replicate to NNMi**

- 1 Determine whether the rate correlator is in use for NNM.
- 2 In the NNMi console, from the **Configuration** workspace, select **Management Event Configurations** (under **Incidents**).
- 3 Open the incident type to be counted.
- 4 On the **Rate** tab, do the following:
  - a Select **Enabled** to enable monitoring.
  - b Set the count window.
  - c Set the time window (**Hours**, **Minutes**, and **Seconds** fields).
  - d Select a management event incident type (for example, **RateCorrelation**) as the new parent event (**Correlation Incident Config**).
  - e Define the **Comparison Criteria**.

For more information, see *Configure Rate (Time Period and Count) for a Management Event Incident* in the NNMi help.

## Event Correlation: Pairwise Cancellation

NNMi does not limit cancellation to a specific time window.


Gather from NNM

1 Determine whether the PairWise correlation is in use in NNM.

2 Determine whether the Transient correlator is in use in NNM.

Replicate to NNMi

3 In the NNMi console, from the **Configuration** workspace, select **Pairwise Configurations** (under **Incidents**).

4 On the **Pairwise Configurations** table, select an existing pair, or click  **New**.

5 Configure the paired event identifiers and the matching criteria.

For more information, see *Pairwise Configuration Form* in the NNMi help.

## Event Correlation: Scheduled Maintenance

NNMi can suppress the monitoring of unavailable nodes. To do so, use the OUT OF SERVICE mode. Unlike NNM, you cannot schedule OUT OF SERVICE maintenance in advance, and you must manually return the objects to MANAGED mode.



SNMP traps sent by devices in OUT OF SERVICE mode are suppressed in NNMi.

If your organization has been using the Scheduled Maintenance correlation, you can use the list of systems that are taken offline together.

Gather from NNM

1 Determine whether the ScheduledMaintenance correlation is in use in NNM.

Replicate to NNMi

2 In the NNMi console, select **Node Groups** from the **Configuration** workspace.

3 Create a node group for each set of nodes in the **NNM Maintenance List**. Set the node groups to be available as view filters.

4 When it is time for maintenance, in the NNMi console select **Nodes** from the **Inventory** workspace.

5 Filter the view to a specific node group by using the **Set node group filter** selector at the top.

6 Select all nodes, and then select **Actions > Management Mode > Out of Service**.

7 After maintenance is completed, select the nodes, and then select **Actions > Management Mode > Manage**.

---

## Phase 6: Upgrade Graphical Visualization (OVW)

In NNM, an OVW map consists of multiple submaps, each of which shows a location or subnet in the network hierarchy. The NNM administrator can define multiple OVW maps and assign a different OVW map to each user.

In NNMi, topology maps are based on the defined node groups. While some topology maps might have a hierarchical relationship, such hierarchy is not limited to network subnets and locations. Additionally, all users can access all available topology maps.

The NNMi upgrade tools can replicate into NNMi the location submap hierarchy of one OVW map. Because the map structure is very different between the two products, the upgrade tools do not transfer nodes, networks, or leaf node elements from NNM.

**Gather from NNM** Upgrade tool approach

- 1 Ensure that the upgrade tools have been set up as described in [Phase 1: Collect Data from the NNM Management Station](#) on page 19.
- 2 Set or create the PERL5LIB environment variable to the following value:
  - *Windows:* `install_dir\migration\lib`
  - *UNIX:* `/opt/OV/migration/lib`
- 3 Identify and open the NNM map that is most representative of the location hierarchy that you want to use in NNMi.
- 4 In the open map, click **File > Export** to create a map data file with the following name and location:
  - *Windows:* `install_dir\migration\ipmap.out`
  - *UNIX:* `/opt/OV/migration/ipmap.out`
- 5 Change to the following directory:
  - *Windows:* `install_dir\migration\`
  - *UNIX:* `/opt/OV/migration/`

- 6 Process the map data file:
  - *Windows:*

```
install_dir\migration\bin\nmmmapmigration.ovpl ipmap.out
```
  - *UNIX:*

```
/opt/OV/migration/bin/nmmmapmigration.ovpl ipmap.out
```

This command creates the `nnmnodegrouplist.csv` and `backgrounds.tar` files, which are available in the following location:

- *Windows:* `install_dir\migration\\OVW.MAPS`
- *UNIX:* `/opt/OV/migration/<hostname>/OVW.MAPS`

**Replicate in NNMi** Upgrade tool approach

- 1 If you have not already done so, copy the `nnmnodegrouplist.csv` and `backgrounds.tar` files from the NNM management server to the following location:
  - *Windows:* `%NnmDataDir%\tmp\migration\\MAPS\`
  - *UNIX:* `$NnmDataDir/tmp/migration/<hostname>/MAPS/`
- 2 Change to the following directory:
  - *Windows:* `%NnmDataDir%\tmp\migration\\MAPS\`
  - *UNIX:* `$NnmDataDir/tmp/migration/<hostname>/MAPS/`

- 3 Import the node group definitions for the NNM location hierarchy into the NNMi database:
  - *Windows:*

```
%NnmInstallDir%\bin\nnmloadnodegroups.ovpl -u <user> \
-p <password> -r false -f nmmnodegrouplist.csv
```
  - *UNIX:*

```
$NnmInstallDir/bin/nnmloadnodegroups.ovpl -u <user> \
-p <password> -r false -f nmmnodegrouplist.csv
```
- 4 Make the NNM background graphics available to NNMi:
  - a Unpack the `backgrounds.tar` file using a tool or command (such as `restoreMigration.ovpl`) that is appropriate for the operating system of the NNMi management server.


- b Copy the extracted files to the following location:

— *Windows:* %NnmDataDir%\shared\nnm\www\htdocs\images\

— *UNIX:* \$NnmDataDir/shared/nnm/www/htdocs/images/

Alternatively, you can transfer the individual image files to the `images` directory using FTP in ASCII mode.

- 5 In the NNMi console, apply the appropriate background graphic to each location node group map:
  - a In the NNMi console, select **Node Groups** from the **Configuration** workspace.
  - b Examine the text in the **Notes** box.
 

If the upgrade tool created the node group, the note field indicates that it was created from an OVW location symbol. If the OVW submap included a background graphic, the note also specifies the image name.
  - c From the **Node Group** form for a replicated node group, click **Actions > Maps > Node Group Map**.
  - d In the map, click **Save Layout**  to create a node group settings object for this node group.
  - e In the same map, click **File > Open Node Group Map Settings**.
  - f On the **Background Image** tab of the **Node Group Map Settings** form, specify the background graphics file that is identified in the note text in the **Node Groups** form for this node group, as described in [step b](#).



On the **Node Group Map Settings** form, the path to the background graphics file is in the following format:

```
/nnmbg/images/<optional_directory_structure>/<filename>
```

In the file system, `/nnmbg/images/` maps to:

— *Windows:* %NnmDataDir%\shared\nnm\www\htdocs\images\

— *UNIX:* \$NnmDataDir/shared/nnm/www/htdocs/images/

(The path in the note text applies to the NNM management station.)

- 6 In the NNMi console, add one or more node groups to the lowest-level topology map in the location hierarchy.

---

## Phase 6: Upgrade Graphical Visualization (Home Base)

In NNM 7.x Advanced Edition, the Home Base can include container views that organize the network topology.

In NNMi, topology maps are based on the defined node groups. While some topology maps might have a hierarchical relationship, such hierarchy is not limited to network subnets and locations. Additionally, all users can access all available topology maps.

The NNMi upgrade tools can replicate into NNMi the Home Base container view hierarchy. Because the map structure is very different between the two products, the upgrade tools do not transfer nodes, networks, or leaf node elements from NNM.

### Gather from NNM Upgrade tool approach

The `nnmmigration.ovpl` tool collected the container view configuration file from the NNM management station.

### Replicate in NNMi Upgrade tool approach

- 1 Change to the following directory:
  - *Windows:* `%NnmDataDir%\tmp\migration\\NNMET\`
  - *UNIX:* `$NnmDataDir/tmp/migration/<hostname>/NNMET/`
- 2 Parse the container view configuration file to create a comma-separated node group list:
  - *Windows:*

```
%NnmInstallDir%\migration\bin\nnmetmapmigration.ovpl \
containers.xml nnmcontainerlist.csv.txt
```
  - *UNIX:*

```
$NnmInstallDir/migration/bin/nnmetmapmigration.ovpl \
containers.xml nnmcontainerlist.csv
```
- 3 Import the node group definitions for the NNM 7.x Advanced Edition Home Base container hierarchy into the NNMi database:
  - *Windows:*

```
%NnmInstallDir%\bin\nnmloadnodegroups.ovpl -u <user> \
-p <password> -r false -f nnmcontainerlist.csv.txt
```
  - *UNIX:*

```
$NnmInstallDir/bin/nnmloadnodegroups.ovpl -u <user> \
-p <password> -r false -f nnmcontainerlist.csv
```
- 4 In the NNMi console, add one or more node groups to the lowest-level topology map in the location hierarchy.

---

## Phase 7: Upgrade Custom Scripts

NNM provides several command-line tools for reading the contents of the NNM databases. These tools can be used from the command line. They can also be incorporated into scripts that were created for your network environment.

On NNMi management servers, the `nmmtopodump.ovpl` command located in the `bin` directory is an enhanced version of what was previously provided as an unsupported tool in the `support` directory. The updated `nmmtopodump.ovpl` command can generate textual output in a format very similar to that of the `NNM ovtodump` command. Additionally, you might be able to replace other NNM commands in custom scripts with the `nmmtopodump.ovpl` command.

Gather from NNM  
Replicate in NNMi

- 1 Copy all custom scripts for reading the NNM databases to a working directory.
- 2 Copy the working directory to the NNMi management server.
- 3 Examine each script for calls to any of the following commands:
  - `ovtodump`
  - `ovobjprint`
  - `ovet_topodump.ovpl`
  - `ovdwquery`
- 4 As appropriate, update each script to call the `nmmtopodump.ovpl` command in place of the commands named in the previous step.



The `nmmtopodump.ovpl` command is not a direct replacement for any of the NNM commands. Compare the `nmmtopodump.ovpl` output with the expected output, and modify each script as needed.

- 5 Test and revise each updated script until it produces the desired results.

For more information, see the `nmmtopodump.ovpl` reference page, or the UNIX manpage.

---

## Upgrade Tools Reference

This section describes the tools that NNMi provides to assist with replicating an NNM 6.x or 7.x configuration to NNMi. This information is current for the product and patch version indicated in the footer of this document.

### Data Collection Tools

Run the data collection tools on the NNM 6.x/7.x management station to gather the NNM configuration information into one place. The procedures for using these tools are described earlier in this chapter.

The data collection tools are delivered with NNMi as two archive files (`migration.zip` for Windows operating systems, `migration.tar` for UNIX operating systems). After NNMi installation, the archive files are available in the following location:

- *Windows*: `%NnmInstallDir%\migration\`
- *UNIX*: `$NnmInstallDir/migration/`

The data collection tools are limited by the availability of commands on the NNM management station. In some cases, these tools will not run to successful completion. If a wrapper script fails, you can run the tools individually. If a single tool fails, you can replicate the intent of the tool (as described here) to collect the data yourself.

[Table 2](#) lists the tools that are included in the data collection tools archive files.

**Table 2 Upgrade Data Collection Tools**

<b>Tool</b>	<b>Description</b>
createMigrationDirs.ovpl	Creates the directory structure to hold the upgrade data that will be collected from the NNM management station. For more information, see <a href="#">NNM Configuration Data Files</a> on page 56.
nmmigration.ovpl	Collects the NNM configuration data. This tool is a wrapper script that runs most of the other tools described in this table.
archiveMigration.ovpl	Packs the collected data into a tar archive file ( <code>&lt;hostname&gt;.tar</code> ) for easy transfer to the NNMi management server.
captureLocale.ovpl	Determines the locale of the NNM management server so that the tools collect the correct version of localized configuration files.
hostnolookup.ovpl	Runs <code>snmpnolookupconf -dumpCache</code> to create a text file ( <code>hostnolookup.conf</code> in the DNS directory) of the hostnames that NNM discovery ignores.
nmmtopodump.ovpl	Runs <code>ovtopodump -lr</code> to create a text file ( <code>ovtopodump.out</code> in the TOPO directory) snapshot of the topology database. This tool is different from the tool of the same name that is installed into the <code>bin</code> directory on the NNMi management server.
ovmapdump.ovpl	Runs <code>ovmapdump -l</code> for each OVW map to create a text file (in the MAPS directory) snapshot of that map database.
ovmibmigration.ovpl	Verifies that all MIBs defined in the NNM <code>snmpmib</code> file have been loaded into NNM.
ovwdbDump.ovpl	Runs <code>ovobjprint</code> to create a text file ( <code>ovobjprint.out</code> in the OVWDB directory) snapshot of the object database that future upgrade tools might use.
snmpCapture.ovpl	Runs <code>xnmsnmconf -dumpCache</code> to create a text file ( <code>snmpCapture.out</code> in the SNMP directory) snapshot of the SNMP configuration database. This tool is different from the tool of the same name that is described in <a href="#">Table 4</a> .

**Table 2 Upgrade Data Collection Tools (cont'd)**

<b>Tool</b>	<b>Description</b>
trapdConfNodes.ovpl	Parses the trapd.conf file to create node lists (EVENTS\NODES\*) that future upgrade tools might use.
nmmmapmigration.ovpl	Parses the export file for an OVW map to identify node groups of the locations in that map (nmmnodegroupelist.csv in the MAPS directory) and to collect the background image files that are used on location submaps (backgrounds.tar in the MAPS directory). Run this command separately from the nmmmigration.ovpl wrapper script.

## NNM Configuration Data Files

The data collection tools store files in the following location:

- *Windows:* `install_dir\migration\\`
- *UNIX:* `/opt/OV/migration//`

Where `<hostname>` is the hostname of the NNM management station. [Table 3](#) lists the contents of the `<hostname>` directory.

**Table 3 File Structure of the Collected NNM Configuration Data**

<b>Directory</b>	<b>Contents</b>
CONFIG	A copy of the NNM CONF directory
DNS	hostnolookup.conf
EVENTS	All trapd.conf files in the NNM configuration Node lists
MAPS	Application registration files Symbol registration files A flat file of each map database
NNMET	(NNM 7.x Advanced Edition) containers.xml
OVW.MAPS	Output of the nmmmapmigration.ovpl tool
OVWDB	A flat file of the object database Field registration files
SNMP	Community strings
TOPO	A flat file of the topology database
WWW	The NNM web interface files



## Data Import Tools for Upgrading

[Table 4](#) lists the tools that NNMi provides for importing NNM 6.x/7.x data into the NNMi database. The upgrade process also uses standard NNMi tools. For information about the standard tools, see the appropriate reference pages, or the UNIX manpages.

**Table 4 Data Import Tools**

Tool	Description
restoreMigration.ovpl	Unpacks the NNM configuration archive created by archiveMigration.ovpl on the NNM 6.x/7.x management station.
nnmetmapmigration.ovpl	Parses the NNM 7.x Advanced Edition Home Base container view definition file (containers.xml) to identify node groups of the locations in that view for NNMi.
nmmibmigration.ovpl	Runs nmmincidentcfg.ovpl to import the MIBs in the NNM snmpmib file into the NNMi database. This tool does not re-load any MIBs that are already loaded in NNMi.
nnmtrapdload.ovpl	Loads trap definitions from the NNM trapd.conf file into the NNMi database. This tool loads only the first definition that it encounters for each trap. It does not re-load any trap definitions that are already loaded in NNMi.
nnmtrapdMerge.ovpl	Merges all non commented lines in the NNM ovtrapd.conf file into the NNMi nnmtrapd.conf file.
snmpCapture.ovpl	Outputs the contents of the snmpCapture.out file to STDOUT, one community string per line. This tool is different from the tool of the same name that is described in <a href="#">Table 2</a> .



# Integrating NNM 6.x or NNM 7.x with NNMi

You can integrate the following HP Network Node Manager (NNM) 6.x or 7.x functionality with HP Network Node Manager i Software (NNMi):

- You can forward events from NNM 6.x or 7.x to the NNMi management server to use the NNMi incident views for managing incident life cycle.
- You can open some NNM 6.x or 7.x views from the NNMi management server.

This integration is useful for controlling the rate of upgrading to NNMi.

This integration is also useful for large managed environments with many NNM 6.x or 7.x management stations. If you do not need the new functionality in NNMi throughout the network, you can maintain a few NNM 6.x or 7.x management stations while using NNMi as your primary network management tool.

You can also use the information in this chapter to integrate a third-party product with NNMi. That product must be able to generate SNMP v1, v2c, or v3 traps and send them to the NNMi management server.

This chapter contains the following topics:

- [Configure Event Forwarding](#) on page 60
- [Configure Remote View Launching](#) on page 64
- [Test the Integration](#) on page 67
- [Troubleshoot Event Forwarding](#) on page 70

# Configure Event Forwarding

To set up event forwarding from the NNM 6.x or 7.x management station to an NNMi management server, complete the following procedures in order:

- [Step 1: Configure NNM 6.x or 7.x to Forward Events to the NNMi Management Server](#)
- [Step 2: \(Optional\) Use Node Level Filtering to Further Reduce Events](#)
- [Step 3: Add the NNM 6.x or 7.x Management Station to the NNMi Topology](#)
- [Step 4: \(Optional\) Save the Management Station Configuration](#)
- [Step 5: Verify NNM 6.x or 7.x Incident Configuration in the NNMi Console](#)

## Step 1: Configure NNM 6.x or 7.x to Forward Events to the NNMi Management Server

On the NNM 6.x or 7.x management station, configure each event that you want forwarded to the NNMi management server. Most of these events will be under the OpenView Enterprise. Interesting events include:

- OV\_Node\_Down (OV\_Node\_Up, Ov\_Node\_Unknown, and so forth)
- OV\_APA\_NODE\_DOWN (OV\_APA\_NODE\_Intermittent, and so forth)
- OV\_Station\_Critical (OV\_Station\_Normal, and so forth)
- OV\_Error (OV\_Warning, OV\_Inform) information about system health
- OV\_Message (OV\_Popup\_Message, and so forth)

For the complete list of recommended NNM 6.x or 7.x events to forward, see the events listed in the **Remote NNM 6.x/7.x Event Configurations** table view in the NNMi console.

### Recommended and Supported Procedure: Use the Event Configuration Window

▶ If you do not have an XServer, see [Alternative Procedure: Manually Edit trapd.conf](#) on page 62.

To configure an NNM 6.x or 7.x event to forward to the NNMi management server, follow these steps:

- 1 At the command prompt, enter:

```
ovw
```

▶ Alternatively, run `xnmtrap` from the command line, and then continue with [step 3](#).

- 2 Click **Options > Event Configuration**.

- 3 In the **Event Configuration** window, select the **Openview** enterprise in the top pane, and then double-click an event name in the bottom pane.

▶ To sort the events by name, click **View > Sort > Event Name**.

Best practice

- 4 Specify the NNMi management server to receive the forwarded events.

If you have created a destination list file, enter the complete path to this file in the **Destination** field. For information about the destination list file format, see [Optional: Destination List File](#) on page 61.

- *Windows*: On the **Forwarding** tab in the **Modify Events** window, enter the host name of the NNMi management server in the **Destination** field.  
Click **Add**, and then click **OK**.
- *UNIX*: In the **Destination** field at the bottom of the **Event Configuration** window, enter the host name of the NNMi management server.



If you do not see the **Destination** field, select the **Forward Event** option in the center of the window.

Click **Add**, and then click **OK**.

- 5 Repeat [step 3](#) and [step 4](#) until all events you want to forward to an NNMi management server are configured.

- 6 Click **File > Save**.

NNM 6.x or 7.x saves the changes to the event configurations and automatically re-reads the new event configuration.

### Optional: Destination List File

If you want to forward several events to the same group of NNMi management servers, you can create a file that lists the forward destinations.

The recommended location for the destination list file is:

- *Windows*: %OV\_CONF%\nmm8EventForwardDestinations.txt
- *UNIX*: \$OV\_CONF/nmm8EventForwardDestinations.txt

The destination list file is a text file with the following format:

- Each line is either one node name or a comment line.
- The first character of a comment line is the # character.

For example:

```
# List of destination NNMi Management Servers to receive events.
# This list should be small enough that it does not overwhelm the NNMi operators.
# In general, the events should be node-related, so that Neighbor Views launched remotely
# from the NNMi management server are meaningful.
#
system1.domain.com
system2.comain.com
system3.domain.com
```

For more information, see the `trapd.conf` manpage.



After creating or changing the destination list file, run the following command to re-read it:

```
xnmevents -event
```

## Alternative Procedure: Manually Edit trapd.conf

If you do not have an XServer, you can manually edit the FORWARD field for each event in the following file:

- Windows: %OV\_CONF%\C\trapd.conf
- UNIX: \$OV\_CONF/C/trapd.conf

Specify either a single NNMi management server or a destination list file. For example:

```
EVENT OV_Message .1.3.6.1.4.1.11.2.17.1.0.58916872 "Application Alert Alarms" Normal
FORMAT $3
FORWARD NNM8Server.domain.com
```

The FORWARD field might also include a list of the remote managers. For example:

```
FORWARD %REMOTE MANAGERS_LIST% /etc/opt/OV/share/conf/nnm8EventForwardDestinations.txt
```



After editing the trapd.conf file, run the following command to force NNM to re-read the event configuration:

```
xnmevents -event
```

## Step 2: (Optional) Use Node Level Filtering to Further Reduce Events

In NNM 7.x, you can configure a node list for certain events. When a node list is present, an event coming into the NNM 7.x management station matches an event configuration only if the event source is in the node list. Thus, an event will be forwarded to the NNMi management server only if the event source is in the node list. A typical use case for a node list is to forward only specific events from important nodes to the NNMi management server.

For information about creating a node list in NNM 7.x, see the information about the sources\_list in the ovtrapd.conf manpage.

## Step 3: Add the NNM 6.x or 7.x Management Station to the NNMi Topology

Include the NNM 6.x or 7.x management station in the NNMi topology so that the NNMi management server receives an incident if the NNM 6.x or 7.x management station goes down.

If the NNM 6.x or 7.x management station is not already in the NNMi **Nodes** inventory view, add the management station to the discovery seeds, and then wait for it to be discovered.

For information about how to add a node to the discovery seeds, see *Discovering Your Network* in the NNMi help.

## Step 4: (Optional) Save the Management Station Configuration

To save the new configuration, run the following command:

```
nnmconfigexport.ovpl -u <user> -p <password> -c station \  
-f <filename>
```

You can later import the backup by running the following command:

```
nnmconfigimport.ovpl -u <user> -p <password> -f <filename>
```

For information about these commands, see their respective reference pages, or the UNIX manpages.

## Step 5: Verify NNM 6.x or 7.x Incident Configuration in the NNMi Console

Verify that the events you forwarded from NNM 6.x or 7.x are configured (as incidents) in NNMi.

To view the NNMi default incident configurations, in the NNMi console, from the **Configuration** workspace, select **Remote NNM 6.x/7.x Event Configurations** (under **Incidents**). This table displays the default incident configurations.

The **Incident** form for this incident type shows the **Origin of NNM 6.x/7.x**.

If one or more of the events that you configured for forwarding from the NNM 6.x or 7.x management station is not listed in the **Remote NNM 6.x or 7.x Events** table, add a new incident configuration for each missing event. For more information, see *Configuring Incidents* in the NNMi help.



The incident categories in NNM 6.x or 7.x are different from those in NNMi. For information about the relationship between the NNM 6.x or 7.x alarm categories and the NNMi incident categories, see [Mapping Categories](#) on page 63.

### Mapping Categories

In NNM 6.x or 7.x, the pre-configured alarm categories are as follows:

- Error Alarms
- Threshold Alarms
- Status Alarms
- Configuration Alarms
- Application Alert Alarms

In NNMi, the pre-configured incident categories are as follows:

- Accounting
- Application Status
- Configuration
- Fault
- Performance
- Security
- Status

Table 5 lists the mapping of NNM 6.x or 7.x alarm categories to NNMi incident categories that HP suggests:

**Table 5 Suggested Category Mappings**

NNM 6.x or 7.x Alarm Category	NNMi Incident Category
Error Alarms	Application Status
Threshold Alarms	Performance
Status Alarms	Status
Configuration Alarms	Configuration
Application Alert Alarms	Application Status

## Configure Remote View Launching

To set up the NNMi management server to display NNM 6.x or 7.x views on the NNMi management server, complete the following procedures in order:

- [Step 1: Install Java Plug-in](#)
- [Step 2: Create an NNM 6.x or 7.x Management Station Entity in NNMi](#)
- [Step 3: \(Optional\) Configure Additional NNM 6.x or 7.x Views](#)

### Step 1: Install Java Plug-in

Although NNMi does not have any requirements for a Java Plug-in, NNM 6.x or 7.x views require the use of a specific version of the Java Plug-in, which depends on the NNM version and operating system.

Review the latest release notes for your version of NNM, and then download and install the correct Java Plug-in version to all web browsers from which NNMi console users will launch NNM Dynamic Views.




## Step 2: Create an NNM 6.x or 7.x Management Station Entity in NNMi

Configure the NNMi management server to associate events received from the NNM 6.x or 7.x management station to an entity in NNMi. This configuration enables the launching of NNM 6.x or 7.x Dynamic Views from the NNMi management server. For example, you can select a Node Down from My7xSystem that is displayed in NNMi, and then launch the URLs back to My7xSystem.



It is important to use the Primary Address that matches the address that is encoded in the event sent by the NNM 6.x or 7.x management station. If you are not sure about this address, look at the **RemoteSenderAddress** in the custom incident attributes for an incident that was forwarded from the NNM 6.x or 7.x management station.

To set up an NNM 6.x or 7.x management station configuration in NNMi, follow these steps:

- 1 In the NNMi console, select **Management Stations (6.x/7.x)** from the **Configuration** workspace.
- 2 Click  **New**.
- 3 On the **Management Station** form, enter the following information:
  - **Name**—An identifier for the NNM 6.x or 7.x management station represented by this configuration.
  - **NNM Version**—The NNM version (6.x or 7.x) of the management station that you are configuring.
  - **IP Address**—An IP address for the NNM 6.x or 7.x management station. This IP address must be reachable from the NNMi management server. You can find the IP address in either of the following ways:
    - Run `ovaddr` at the command line on the NNM 6.x or 7.x management station.
    - Determine the custom incident attribute (CIA) of an incident that has been forwarded from the NNM 6.x or 7.x management station.




This method works only if you have already completed the procedures that are described in [Configure Event Forwarding](#) on page 60 and if a configured event has been generated on the NNM 6.x or 7.x management station and forwarded to the NNMi management server.



- **ovas Port**—The port number of the OpenView Application Server (ovas) for the NNM 7.x management station that you are configuring. On NNM 7.x management stations, the port number is usually 7510.

The ovas port also applies to NNM 6.x with the Extended Topology add-on.

- **Web Server Port**—The port number of the web server for the NNM 6.x or 7.x management station that you are configuring:
    - For NNM 6.x management stations on a Windows operating system, this port number is usually 80.
    - For NNM 6.x management stations on a UNIX operating system, this port number is usually 3443.
    - For NNM 7.x management stations on all operating systems, this port number is usually 3443.
  - **Description**—A description of the NNM 6.x or 7.x management station that you are configuring.
- 4 Click  **Save and Close**.
  - 5 Sign out of the NNMi console.

The next time you sign in to the NNMi console, the **Actions** menu will contain new items for launching NNM 6.x or 7.x views.

### Step 3: (Optional) Configure Additional NNM 6.x or 7.x Views

The following URLs are not added out-of-the-box. You can add any of these URLs to the NNM 6.x or 7.x deployment.

#### URLs That Do Not Require a Selection

- **MIB Browser Example URL:**

`http://192.168.1.xxx:3443/OvCgi/OpenView5.exe?Action=Snmp&Host=speed2.cnd.hp.com`

- **Report Presenter Example URL:**

`http://192.168.1.xxx:3443/OvCgi/nmRptPresenter.exe`

- **Topology Summary Example URL:**

`http://192.168.1.xxx:7510/topology/summary`

- **SNMP Data Presenter (MIB Form/Table contrib. graphs):**

`http://192.168.1.xxx:3443/OvCgi/snmpviewer.exe?Context=Performance&sel=10.97.245.242`

- **OV Launcher Example URL:**

`http://system.example.com:3443/OvCgi/ovlaunch.exe`

- **jovw Example URL:**

(Web-based ovw, requires an ovw session running; otherwise, you see the error message "Cannot find an ovw on host ..." with map named default using sessionID xxxx:x):

`http://system.example.com:3443/OvCgi/jovw.exe`



This URL can take a context node and map name, with option such as:

`jovw.exe?mapName=default&ObjectName=10.1.12.33`

- **ovalarm Example URL:**

`http://system.example.com:3443/OvCgi/ovalarm.exe`

- Form to request topology details (type in a node by Name, IP Address, Physical Address UUID, OvwId):

`http://192.168.1.xxx:7510/topology/topoDetail`

### URLs That Require a Selection

- Node Details using an ovwId:

`http://192.168.1.xxx:7510/topology/topoDetail?objectType=ovwId&objectValue=3&Show+Details=Show+Details`

- Node Details using a UUID:

`http://192.168.1.xxx:7510/topology/topoDetail?objectType=uuid&objectValue=3dasfasdf&Show+Details=Show+Details`

---

## Test the Integration

To verify that you have correctly set up the NNM 6.x or 7.x integration with the NNMi management server, complete one or both of the following procedures, as appropriate:

- [Test 1: Verify Event Forwarding](#)
- [Test 2: Launch NNM 6.x or 7.x Dynamic Views from NNMi](#)

### Test 1: Verify Event Forwarding

Under normal network conditions, NNM 6.x or 7.x generally receives network events. The NNM 6.x or 7.x management stations forwards the configured events to NNMi, which displays them as remotely generated 6.x/7.x incidents. To expedite testing, you can generate a test event, or you can create an actual network failure on a test network or test device.

To verify event forwarding from the NNM 6.x or 7.x management station to the NNMi management server, follow these steps:

- 1 On the NNM 6.x or 7.x management station, create a situation that generates one of the forwarded events.

The simplest approach is to run the `sendMsg.ovpl` command on the NNM 6.x or 7.x management station. For information about how to run this command, see [sendMsg.ovpl](#) on page 69.

Another approach is to generate or simulate a network fault on the NNM 6.x or 7.x system. See [Generate Test Interface Down and Interface Up Events](#) on page 68.

- 2 View the generated event in the NNMi console by selecting **NNM 6.x or 7.x Events** from the **Incident Browsing** workspace.

The event that you generated from the NNM 6.x or 7.x management station should be visible in this view.



Alternatively, you can run `nnmdumpevents -t` on the NNMi management server to see the list of events that the NNMi management server has received.

## Generate Test Interface Down and Interface Up Events



The following test procedure requires changes to the NNM 6.x or 7.x configuration. Do not perform this procedure on a production network management station.

- 1 On an NNM 7.x management station, disable Extended Topology if it is enabled:

```
setupExtTopo.ovpl -disable
```

- 2 On the NNM 6.x or 7.x management station, in the ECS user interface, note which correlations are active, and then disable all correlations.
- 3 Generate test interface down events, which might also cause a node down event, by running the following command once for each IP interface on the node:

```
ovtopofix -S Down <IPADDR>
```

Where *<IPADDR>* is the IP address of one of the interfaces in the NNM 6.x or 7.x management station topology. To determine the IP addresses to use, run the following command:

```
ovtopodump > topology.txt
```

In the *topology.txt* file, search for the word *NODES*, and then locate the entries for the NNM 6.x or 7.x management stations. For example:

```

NODES:
1516          IP      mplscexx.xxx.xx.com   Marginal   10.2.120.72
1516/1517     IP      mplscexx.xxx.xx.com   Normal     10.2.120.72
1516/2046     IP      mplscexx.xxx.xx.com   Critical    10.97.255.28
1516/2047     IP      mplscexx.xxx.xx.com   Critical    10.16.160.5
1516/2050     -       mplscexx.xxx.xx.com   Normal     -
1516/2051     -       mplscexx.xxx.xx.com   Normal     -
1516/2052     -       mplscexx.xxx.xx.com   Normal     -
1516/2053     -       mplscexx.xxx.xx.com   Normal     -
1516/5250     IP      mplscexx.xxx.xx.com   Critical    10.40.40.1
1516/5251     IP      mplscexx.xxx.xx.com   Critical    10.40.40.2

```

When all IP interfaces have status *Critical*, NNM shows the node as down.



Alternatively, you can specify the node name or the topology ID for the NNM 6.x or 7.x management station as the last argument to the *ovtopofix* command. For other options, see the *ovtopofix* manpage.



Make sure that the events you are testing (in this case, *OV\_IF\_Up/OV\_IF\_Down*, which are *.1.3.6.1.4.1.11.2.17.1.0.58916866* and *.1.3.6.1.4.1.11.2.17.1.0.58916867*, respectively) are configured to be forwarded to the NNMi management server.

- 4 To clean up the events browser, run the following command once for each IP interface to generate Interface Up and Node Up events:

```
ovtopofix -S Up <IPADDR>
```

- 5 On the NNM 6.x or 7.x management station, in the ECS user interface, re-enable the correlations that you disabled in [step 2](#).
- 6 If you disabled Extended Topology in [step 1](#), re-enable it on the NNM 7.x management station:

```
setupExtTopo.ovpl
```

## sendMsg.ovpl

You can run the `sendMsg.ovpl` command to generate an `OV_Message` event. For example:

- *Windows:*

```
%OV_CONTRIB%\NNM\sendMsg\sendMsg.ovpl "" "Test from %COMPUTERNAME%"
```

- *UNIX:*

```
$OV_CONTRIB/NNM/sendMsg/sendMsg.ovpl "" "Test from `hostname` on `date`"
```

Each time you run the `sendMsg.ovpl` command, NNM 6.x or 7.x generates an `OV_Message` event containing the text that you included in the `sendMsg.ovpl` command line. For example:

```
1183160690 6 Fri Jun 29 17:44:50 2007 <none> a Test from speed2 on
Fri Jun 29 17:44:50 MDT 2007;1 17.1.0.58916872 0
```

This event is visible in the **All Alarms** browser on the NNM 6.x or 7.x management station.

### Best practice

To facilitate identification of new alarms, delete all of the alarms in the **All Alarms** browser before running the `sendMsg.ovpl` command.

## Test with Traps to NNM 6.x or 7.x System

If you configured NNM 6.x or 7.x to forward traps, you should see received traps that are being forwarded.

You can manually generate traps on the NNM 6.x or 7.x management station with a command similar to the following example:

```
snmptrap -p 162 hostname "" "" 6 1234 "" .1.3.6.1.3.1.1.5.3 \
octetstring "Test Trap"
```



The example generates an `SNMP_Link_Down` trap. Use the event object identifier for a trap that you configured to be forwarded.

`hostname` is the name of the NNM 6.x or 7.x system. For more information, see the `snmptrap` manpage.

## Test 2: Launch NNM 6.x or 7.x Dynamic Views from NNMi

- 1 In the NNMi console, open the NNM 6.x or 7.x management station that you configured.

The following actions are available on the **Actions** menu:

- NNM 6.x or 7.x Home Base
- NNM 6.x or 7.x ovw
- NNM 6.x or 7.x MIB Browser
- NNM 6.x or 7.x Launcher
- NNM 6.x or 7.x Alarms



If these actions are not available, sign out of the NNMi console, and then sign in to the NNMi console again.

- 2 Open each of the views from the **Actions** menu.

## Troubleshoot Event Forwarding

If you did not see the expected NNM 6.x or 7.x events in the **NNM 6.x or 7.x Events** incident view, follow these steps to troubleshoot the problem:

- 1 On the NNM 6.x or 7.x management station, run the following command:

```
ovdumpevents -t -l <n>
```

Where *<n>* specifies the number of minutes to go back in the event history. For example, when the value for *n* is 1, the `ovdumpevents` command displays the events that have been generated on the NNM 6.x or 7.x management station in the last minute.

- 2 If an expected event is not included in the `ovdumpevents` output, the event was not generated. See the NNM 6.x or 7.x documentation for information about troubleshooting this situation.
- 3 Repeat [step 1](#) until all expected events are included in the `ovdumpevents` output on the NNM 6.x or 7.x management station.
- 4 On the NNMi management server, run the following command:

```
nnmdumpevents -t -l <n>
```

Where *<n>* specifies the number of minutes to go back in the event history. For example, when the value for *n* is 1, the `nnmdumpevents` command displays the events that have been generated on the NNMi management server in the last minute.

- 5 For each expected event that is not included in the `nnmdumpevents` output, verify the configuration of that event in the **Event Configurator** window on the NNM 6.x or 7.x management station.
  - Verify that the **Forward Event** option is selected.
  - Verify the names or IP addresses of the NNMi management servers in the **Forwarded Event Destinations** list.

For more information, see [Step 1: Configure NNM 6.x or 7.x to Forward Events to the NNMi Management Server](#) on page 60.

- 6 Repeat [step 5](#) until all expected events are included in the `nnmdumpevents` output on the NNMi management server.
- 7 In the NNMi console, examine the **NNM 6.x or 7.x Events** incident view.

If the results are not as expected, verify the incident configuration from the **Remote NNM 6.x or 7.x Event Configurations** table.



`nnmtrapconfig.ovpl -dumpBlockList` outputs information about the current incident configuration, including SNMP traps that were not passed into the incident pipeline because of non-existent or disabled incident configurations.

For information about displaying incoming traps that do not have an incident configuration in NNMi, see “Enabling and Configuring Incidents for Undefined Traps” in the *NNMi Deployment Reference*.

# Upgrading from NNMi 8.0x or 8.1x



For information about upgrading from NNM 6.x or 7.x to NNMi 9.20, see the [Upgrading from 6.x or 7.x](#) on page 13.

To upgrade from NNMi 8.0x to NNMi 9.20, you must first upgrade to NNMi 8.1x.

To upgrade from NNMi 8.1x to NNMi 9.20, you must first upgrade to NNMi 9.0x.

You can upgrade NNMi according to the information shown in [Table 6](#). For best results, upgrade to NNMi 8.1x patch 8 or newer before upgrading to NNMi 9.0x. The information shown in [Table 6](#) assumes you have NNMi 8.10 or newer installed on the NNMi management server.

**Table 6 Supported NNMi Upgrades**

NNMi Version	Upgrade to NNMi 9.0x
8.10	Supported
8.1x Patch 1 or newer	Supported

If you plan to upgrade an earlier version of NNMi 8.1x that is running in an NNMi application failover or HA (High Availability) configuration, the supported upgrade path is to temporarily unconfigure HA or application failover, upgrade the NNMi management server to NNMi 9.00, then reconfigure HA or application failover. For detailed information, see the [Upgrading NNMi under HA from NNMi 8.1x to NNMi 9.01](#) on page 103.

See [Table 7](#) to view the supported upgrade paths to NNMi 8.10. If you have a version of NNM older than NNMi 8.10 installed, you cannot upgrade directly to NNMi 9.0x Patch 4.

**Table 7 Supported NNMi Upgrades (to NNMi 8.10)**

Current Version	Upgrade to NNMi 8.02	Upgrade to NNMi 8.03 or higher *	Upgrade to NNMi 8.10
NNMi 8.01	Supported	Supported	Install NNMi version 8.10.
NNMi 8.02	NA	Supported	Install NNMi version 8.10.
NNMi 8.03 or higher *	NA	NA	Install NNMi version 8.10.

\* Excluding NNMi 8.1x. To install NNMi patches, see the patch installation instructions.

There are several upgrade scenarios you could encounter. This section contains the following chapters:

- [Upgrading the NNMi Management Server in Place from 8.0x or 8.1x](#), which describes the following upgrade scenario:
  - Upgrading from NNMi 8.0x to NNMi 8.1x or NNMi 8.1x to NNMi 9.00 on the same hardware and operating system.
- [Upgrading to a Different NNMi Management Server from 8.0x or 8.1x](#), which describes the following upgrade scenario:
  - Upgrading from NNMi 8.0x to NNMi 8.1x or NNMi 8.1x to NNMi 9.00 on the same version operating system.
- [Moving NNMi from Red Hat Linux 4.6 to 5.2 or 5.3](#). NNMi 9.00 does not support Red Hat Linux 4.6. You must change the operating system to Red Hat Linux 5.2 or 5.3 before migrating to NNMi 9.00.
- [Migrating NNMi Oracle Data](#). Explains the steps to take to move the Oracle data used by your NNMi management server from one Oracle database instance to another.
- [Additional Upgrade Information](#). Explains some areas that NNMi 9.0x differs from earlier versions of NNMi.



# Upgrading the NNMi Management Server in Place from 8.0x or 8.1x

This chapter describes the process for upgrading an existing NNMi management server to NNMi 9.0x.

This chapter contains the following topics:

- [Start from NNMi 8.0x](#)
- [Upgrade an Existing NNMi Management Server to NNMi 9.0x](#)

---

## Start from NNMi 8.0x

Upgrade the NNMi management server to version 8.10 or later. Continue with the instructions shown in [Upgrade an Existing NNMi Management Server to NNMi 9.0x](#) on page 73.

---

## Upgrade an Existing NNMi Management Server to NNMi 9.0x

Read the NNMi 9.20 *Preinstallation Checklist* chapter in the *HP Network Node Manager i Software Installation Guide* and [Additional Upgrade Information](#) on page 95 before continuing. There are notable changes to the *NNMi Installation Guide*. For example, if you use an Oracle database instance instead of the embedded database, you should set the FLASHBACK ANY TABLE permission, as this enables NNMi to create restore points during migration.

The following steps explain how to upgrade an NNMi management server to NNMi 9.0x. The following steps assume you have NNMi 8.10 or later running on the NNMi management server.

- 1 Backup the NNMi management server using the `nnmbackup.ovpl` script. Do this as a precaution, as you would only use this backup in the unlikely event of a failed migration. For more information, see the `nnmbackup.ovpl` reference page, or the UNIX manpage.

- 2 *Oracle Database Only:* If the NNMi management server uses an Oracle database, have your Oracle database administrator back up the NNMi data. As mentioned earlier, have your Oracle database administrator set the FLASHBACK ANY TABLE permission, as this enables NNMi to create restore points during migration
- 3 *Oracle Database Only:* Use the `nmconfigexport.ovpl` script to back up configuration information from the NNMi management server. Do this as a precaution, as you would only use this backup in the unlikely event of a failed migration. For more information, see the `nmconfigexport.ovpl` or `nmconfigimport.ovpl` reference pages, or the UNIX manpages.



Never edit a file exported with the `nmconfigexport.ovpl` script before using the `nmconfigimport.ovpl` script to import the file.

- 4 Install NNMi 9.0 on the NNMi management server using instructions from the *NNMi Installation Guide*.



*Oracle Database Only:* If your Oracle database administrator does not set the FLASHBACK ANY TABLE permission, you will see a warning about that missing permission after the install completes. You can ignore this warning.

- 5 Verify that the information from the NNMi management server migrated successfully.

# Upgrading to a Different NNMi Management Server from 8.0x or 8.1x

This chapter describes the process for upgrading to NNMi version 9.0x on a new system while maintaining the configuration of the existing NNMi management server.

This chapter contains the following topics:

- [Start from NNMi 8.0x](#)
- [Upgrade to a Different NNMi Management Server](#)

---

## Start from NNMi 8.0x

Upgrade the NNMi management server to version 8.10 or later. Continue with the instructions shown in [Upgrade to a Different NNMi Management Server](#) on page 75.

---

## Upgrade to a Different NNMi Management Server

Read the NNMi 8.1x *Preinstallation Checklist* chapter in the *HP Network Node Manager i Software Installation Guide* and [Additional Upgrade Information](#) on page 95 before continuing. There are notable changes to the *HP Network Node Manager i Software Interactive Installation Guide*. For example, if you use an Oracle database instance instead of the embedded database, you should set the FLASHBACK ANY TABLE permission, as this enables NNMi to create restore points during migration.

The following steps explain how to copy data from an existing NNMi management server to a target NNMi management server. The following steps assume you have NNMi 8.10 or higher running on the existing NNMi management server.



If you want to change the Oracle database server, complete that process before or after the upgrade to NNMi 8.1x. For information, see [Migrating NNMi Oracle Data](#) on page 93.

- 1 As a precaution, back up the existing (source) NNMi 8.1x management server using the `nmbbackup.ovpl` script. Label this backup for 8.1x. For more information, see the *nmbbackup.ovpl* reference page, or the UNIX manpage for NNMi 8.1x.
- 2 If the existing (source) NNMi management server uses an Oracle database, have your Oracle database administrator back up the NNMi 8.1x data. As mentioned earlier, have your Oracle database administrator set the FLASHBACK ANY TABLE permission, as this enables NNMi to create restore points during migration.
- 3 Install NNMi 9.00 and the latest consolidated patch (if any) on the source NNMi management server using instructions from the *HP Network Node Manager i Software Installation Guide*.



*Oracle Database Only:* If your Oracle database administrator does not set the FLASHBACK ANY TABLE permission, you will see a warning about that missing permission after the install completes. You can ignore this warning.

- 4 Verify that NNMi 9.0x is working correctly on the source NNMi management server.
- 5 Back up NNMi 9.0x on the source NNMi management server using the `nmbbackup.ovpl` script. Label this backup for NNMi 9.0x. You will need it to copy data to the target NNMi management server. For more information, see the *nmbbackup.ovpl* reference page, or the UNIX manpage for NNMi 9.0x.
- 6 Install NNMi 9.20 and the latest consolidated patch (if any) on the target NNMi management server using instructions from the *HP Network Node Manager i Software Interactive Installation Guide*. To migrate the data from [step 5](#), the target NNMi management server must be running the same operating system version. NNMi does not support data migration to an NNMi management server running on a different operating system.
- 7 Use the `nmmrestore.ovpl` script to copy NNMi database information to the target server. For more information, see the *nmmrestore.ovpl* reference page, or the UNIX manpage.
- 8 Obtain and install a new license on the target NNMi management server.
- 9 Verify that the information from the target NNMi management server migrated successfully from the existing NNMi management server.

# Changing the NNMi Management Server from 8.0x or 8.1x

You can duplicate the HP Network Node Manager i Software configuration on another system, for example, to move from a test environment to a production environment or to change the hardware of the NNMi management server.

You can change the IP address of the NNMi management server without affecting the NNMi configuration.

This chapter contains the following topics:

- [Best Practices for Preparing the NNMi Configuration to Be Moved](#)
- [Moving the NNMi Configuration and Embedded Database](#)
- [Moving the NNMi Configuration](#)
- [Restoring the NNMi Public Key Certificate](#)
- [Changing the IP Address of a Standalone NNMi Management Server](#)
- [Changing the Hostname or Domain Name of an NNMi Management Server](#)
- [Changing the Oracle Database Instance Connection Information](#)
- [Changing the Password that NNMi Uses to Connect to the Oracle Database Instance](#)

---

## Best Practices for Preparing the NNMi Configuration to Be Moved

The following best practices apply to moving the NNMi configuration to a different system:

- If the node group configuration uses hostnames to identify managed nodes, the production and test NNMi management servers must use the same DNS servers. In the case that the production and test systems use different DNS servers, changes in the resolved name for a managed node might result in different polling settings between the two NNMi management servers.

- You can limit the configuration export to a single author. Create a new author value that is unique to your group or company. Specify this author value when you create or modify any of the following items:
  - Device profile
  - Incident configuration
  - URL action
- If you plan to install Smart Plug-ins (iSPIs), see the appropriate NNM iSPI document. Documentation for all NNM iSPIs is available on the HP Software Product Manuals web site at <http://support.openview.hp.com/selfsolve/manuals>.

---

## Moving the NNMi Configuration and Embedded Database

To move the NNMi configuration and the embedded database, for example from a test system to a production system, perform a complete backup of all NNMi data on the source (test) system, and then restore the backup to the target (production) system. To ensure that no changes are made to the NNMi database after the backup is made, stop all NNMi processes and create an offline backup. For example:

```
nmmbackup.ovpl -type offline -scope all \
-target nmi_backups\offline
```

Ensure that the requirements listed in "Different System Restore" in the *NNMi Deployment Reference* are met on the new system, and then run a command similar to the following example:

```
nmrestore.ovpl -source nmi_backups\offline\newest_backup
```



NNMi uses the same SSL certificate for accessing the database (embedded or external) and supporting HTTPS access to the NNMi console. The certificate for accessing the database was created when the NNMi processes first started on the source system. This certificate is included in the backup and restore data. Without this certificate NNMi cannot access the database from the target system.

However, for HTTPS access to the NNMi console, the SSL certificate must be generated on the target system. Because the current implementation of jboss does not support certificate merging, NNMi does not support HTTPS access to the NNMi console on a system that was set up by restoring data from a different system. If the target system must support HTTPS access to the NNMi console, use the procedure described in [Moving the NNMi Configuration](#) on page 79, and then begin data collection fresh on the target system.

## Moving the NNMi Configuration

Use the `nnmconfigexport.ovpl` command to output the NNMi configuration to an XML file. Then, use the `nnmconfigimport.ovpl` command to pull this configuration from the XML file into NNMi on the new system.



Do not edit a file exported with the `nnmconfigexport.ovpl` script before using the `nnmconfigimport.ovpl` script to import the file.

For information about these commands, see the appropriate reference pages, or the UNIX manpages.



The `nnmconfigexport.ovpl` command does not retain SNMPv3 credentials. For more information, see the `nnmconfigexport.ovpl` reference page, or the UNIX manpage.



You can only move the NNMi configuration. HP does not support moving topology or incident data from one NNMi management server to a different NNMi management server. Nor does HP support moving iSPI data, such as performance data that was collected for the NNM iSPI for Metrics.

## Restoring the NNMi Public Key Certificate



If the NNMi management server participates in NNMi application failover or is a member of a high availability (HA) cluster, contact your support representative for assistance.

The `nnm.keystore` file stores the public key certificate that NNMi uses for encryption. The NNMi installation process creates the `nnm.keystore` file and links the certificate in this file to the `nms_sec_key` record in the NNMi database (Postgres or Oracle).

If NNMi is subsequently uninstalled, but the Oracle user and database tables for NNMi are not deleted (cascaded delete of the Oracle user) before a subsequent reinstall, the `nms_sec_key` entry is not valid for the newly created `nnm.keystore` file.

To restore the NNMi public key certificate, complete the following tasks:

- [Task 1: Determine the Status of the KeyManager Service](#)
- [Task 2: Back up the Current `nnm.keystore` File](#)
- [Task 3: Attempt to Locate the Original `nnm.keystore` File](#)
- [Task 4: If Available, Restore the Original `nnm.keystore` File](#)

### Task 1: Determine the Status of the KeyManager Service

- 1 Run the following command:

```
ovstatus -v ovjboss
```

- 2 In the command output, verify that the KeyManager service is not running, which usually indicates that the `nnm.keystore` file is corrupt or missing.

If the `ovstatus` output shows that the KeyManager service is started, contact your support representative for assistance.

## Task 2: Back up the Current nnm.keystore File

- 1 Change to the directory that contains the NNMi trust store:
  - *Windows*: %NnmDataDir%\shared\nnm\certificates
  - *UNIX*: \$NnmDataDir/shared/nnm/certificates
- 2 For backup purposes, save copies of the following files:
  - nnm.keystore
  - nnm.truststore

## Task 3: Attempt to Locate the Original nnm.keystore File

- 1 Determine the fingerprint of the security key in the NNMi database:
  - For the embedded Postgres database, enter the following:
    - *Windows*:  
 %NnmInstallDir%\nonOV\Postgres\bin\psql -U postgres \  
 -d nnm -c "<database\_command>"
    - *UNIX*:  
 \$NnmInstallDir/nonOV/Postgres/bin/psql -U postgres \  
 -d nnm -c "<database\_command>"

Replace <database\_command> with the following SQL command string:

```
select fingerprint from nms_sec_key;
```

- For an Oracle database, ask the Oracle database administrator to run the <database\_command> (described for the embedded database earlier in this step) in the appropriate Oracle administration tool.

The command results should be a single database row. The correct nnm.keystore file also contains this fingerprint.

- 2 Identify a backup nnm.keystore file to test.

This file might be in a backup of the NNMi management server in the original installation directory.

- 3 Test the fingerprint of a backup nnm.keystore file:

- a Change to the directory that contains the NNMi certificates:

- *Windows*: %NnmDataDir%\shared\nnm\certificates

- *UNIX*: \$NnmDataDir/shared/nnm/certificates

- b Examine the contents of the key store:

- *Windows*:

```
%NnmInstallDir%\nonOV\jdk\b\bin\keytool -list \  
-keystore nnm.keystore
```

- *UNIX*:

```
$NnmInstallDir/nonOV/jdk/b/bin/keytool -list \  
-keystore nnm.keystore
```

When prompted for the key store password, enter: **nnmkeypass**

The key store output is of the form:



```
Keystore type: jks
Keystore provider: SUN
Your keystore contains 1 entry
selfsigned, Oct 28, 2008, keyEntry,
Certificate fingerprint (MD5):
29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02
```

- c Compare the value of the MD5 fingerprint from this `nnm.keystore` file with the fingerprint in the NNMi database (from [step 1](#) of this task).
  - If the fingerprints match exactly, you have located a good `nnm.keystore` file for this NNMi database. Continue with [Task 4: If Available, Restore the Original `nnm.keystore` File](#).
  - If the fingerprints do not match exactly, repeat [Task 3: Attempt to Locate the Original `nnm.keystore` File](#).

If you cannot locate the original `nnm.keystore` file using the above procedure, contact your support representative for assistance. Do not continue with [Task 4: If Available, Restore the Original `nnm.keystore` File](#).

#### Task 4: [If Available, Restore the Original `nnm.keystore` File](#)

If you located the correct `nnm.keystore` file, restore that file by following these steps:

- 1 Stop NNMi:  
**ovstop**
- 2 Copy the located `nnm.keystore` file on top of the existing file in the following location:
  - *Windows:* %NnmDataDir%\shared\nnm\certificates
  - *UNIX:* \$NnmDataDir/shared/nnm/certificates
- 3 Start NNMi:  
**ovstart**
- 4 Run the following command:  
**ovstatus -v ovjboss**
- 5 In the command output, verify that the KeyManager service is started.

After you have verified that NNMi is working correctly, you can remove the backup copy of the `nnm.keystore` file from [Task 2: Back up the Current `nnm.keystore` File](#).

---

## Changing the IP Address of a Standalone NNMi Management Server

To change the IP address of the NNMi management server, follow these steps:

- 1 Go to **<http://www.webware.hp.com>**.
- 2 Click **Manage Licenses**.
- 3 Log in; then obtain your new license key by following the procedures to complete the move process.
- 4 Configure the NNMi management server with the new IP address.
- 5 Configure the DNS servers to recognize the new IP address of the NNMi management server.
- 6 Reboot the NNMi management server.
- 7 At a command prompt, enter the following command:  

```
nnmlicense.ovpl NNM -g
```
- 8 In the **Autopass: License Management** dialog box, click **Remove License Key**.
- 9 Select the license key to remove.
- 10 Select **Remove Licenses permanently**.
- 11 Click **Remove**; then close the dialog box.
- 12 Copy the new license key that you obtained in [step 3](#) into a text file named `license.txt`.
- 13 At a command prompt, enter the following command:  

```
nnmlicense.ovpl NNM -f license.txt
```

# Changing the Hostname or Domain Name of an NNMi Management Server



If the NNMi management server participates in NNMi application failover or is a member of a high availability (HA) cluster, contact your support representative for assistance.

To change the hostname, the domain name, or both, of the NNMi management server, complete the following tasks:

- [Task 1: Prepare the System](#)
- [Task 2: Create a New NNMi Public Key Certificate](#)
- [Task 3: Change the Fully-Qualified Domain Name of the NNMi Management Server](#)
- [Task 4: Update the HTTPS Configuration with the New Certificate](#)
- [Task 5: Restart, Update, and Refresh Systems](#)
- [Task 6: Back up NNMi](#)

## Task 1: Prepare the System

- 1 Follow your standard procedure to take a complete NNMi backup.



Clearly label this backup as before changing the name of the NNMi management server.

- 2 Rename the system.

If necessary, reboot the system. The ovjboss process might not start completely.

- 3 If the IP address of the NNMi is also changing, complete the steps in [Changing the IP Address of a Standalone NNMi Management Server](#) on page 82.

- 4 Stop NNMi:

```
ovstop
```

- 5 Change to the directory that contains the NNMi certificates:

- *Windows:* %NnmDataDir%\shared\nnm\certificates
- *UNIX:* \$NnmDataDir/shared/nnm/certificates

- 6 For backup purposes, save copies of the following files:

- nnm.keystore
- nnm.truststore

## Task 2: Create a New NNMi Public Key Certificate

Create a new certificate for this NNMi management server in the nnm.keystore file. The next time that the ovjboss process starts successfully, NNMi updates the database access to use the new certificate.

- 1 Change to the directory that contains the NNMi certificates:

- *Windows:* %NnmDataDir%\shared\nnm\certificates
- *UNIX:* \$NnmDataDir/shared/nnm/certificates

Run all commands in this procedure from the `certificates` directory.

- 2 Generate a new public/private key pair (certificate) in the keystore by running the following command:

- *Windows:*  

```
%NnmInstallDir%\nonOV\jdk\b\bin\keytool -genkey \  
-alias "<unique_alias>" -keyalg rsa \  
-dname "cn=<hostname>, dc=<domain_name_by_parts>" \  
-keypass "nnmkeypass" -validity 36500 \  
-keystore nnm.keystore -storepass "nnmkeypass"
```
- *UNIX:*  

```
$(NnmInstallDir)/nonOV/jdk/b/bin/keytool -genkey \  
-alias "<unique_alias>" -keyalg rsa \  
-dname "cn=<hostname>, dc=<domain_name_by_parts>" \  
-keypass "nnmkeypass" -validity 36500 \  
-keystore nnm.keystore -storepass "nnmkeypass"
```

Replace `<alias>` with a unique value such as the new hostname of the NNMi management server, for example: `newnnmi`

Replace `<hostname>` with the new fully-qualified domain name of the NNMi management server, for example: `newnnmi.servers.example.com`

Replace `dc=<domain_name_by_parts>` with the individual components of the new domain in which the NNMi management server resides. For example, for the NNMi management server `newnnmi.servers.example.com`, specify:  
`dc=servers, dc=example, dc=com`

For more information about the `keytool` command, search for “Key and Certificate Management Tool” at [java.sun.com](http://java.sun.com).

### Task 3: Change the Fully-Qualified Domain Name of the NNMi Management Server

To set NNMi to use the new fully-qualified domain name of the NNMi management server, use the `nnmsetofficialfqdn.ovpl` command. For example:

```
nnmsetofficialfqdn.ovpl newnnmi.servers.example.com
```

For more information, see the `nnmsetofficialfqdn.ovpl` reference page, or the UNIX manpage.

### Task 4: Update the HTTPS Configuration with the New Certificate

Configure the Tomcat server by editing the following file:

```
$jboss.home.dir/server/nms/deploy/jboss-web.deployer/server.xml
```

The default value of `$jboss.home.dir` is as follows:

- *Windows:* `%NnmInstallDir%\nonOV\jboss\nms`
- *UNIX:* `$(NnmInstallDir)/nonOV/jboss/nms`

If the NNMi web server uses the HTTPS protocol, update the HTTPS configuration by following these steps:

- 1 Open the `server.xml` file in any text editor.
- 2 In the uncommented `https` connector block, change the value of the `keyAlias` parameter to match the alias value you used for the new certificate in [Task 2: Create a New NNMi Public Key Certificate](#).
- 3 Save the `server.xml` file.

### Task 5: Restart, Update, and Refresh Systems

1 Start NNMi:

**ovstart**

- 2 Update the connectivity between the NNMi management server and any NNM iSPIs running on dedicated servers to use the new fully-qualified domain name of the NNMi management server.
- 3 Update the connectivity between the NNMi management server and any integrated applications to use the new fully-qualified domain name of the NNMi management server.  
  
If necessary, update the single sign-on configuration for the integrated application to trust the new NNMi certificate.
- 4 If the NNMi database contains any encrypted data (such as SNMPv3 passphrases), this data was encrypted with the old security key. The new security key cannot decrypt the data. Contact your support representative for assistance deleting and recreating these configuration items.

### Task 6: Back up NNMi

Follow your standard procedure to take a complete NNMi backup.



Restoring NNMi from a backup made before changing the name of the NNMi management server overwrites the `nnm.keystore` file, thereby making the NNMi database inaccessible. If you must restore NNMi data from an old backup, contact your support representative for assistance.

## Changing the Oracle Database Instance Connection Information

NNMi can be connected to one Oracle database instance at a time. You can configure this connection.

Reasons to change the Oracle database instance connection information include the following:

- The Oracle database server name must be changed.
- The port for connecting to the database conflicts with another process, or corporate policies require the use of a non-default port.
- The database instance must be renamed (for example, to meet corporate policies).
- The Oracle database server hardware must be changed.

To change the Oracle database instance that NNMi uses, complete the following tasks:

- [Task 1: Update the Oracle Database Instance](#)
- [Task 2: Update the NNMi Configuration](#)

### Task 1: Update the Oracle Database Instance

1 Stop NNMi:

**ovstop**

- 2 Prepare the Oracle database by moving the database, renaming the Oracle database server, or other necessary changes.
- 3 Verify that the target Oracle database instance meets the following prerequisites:
  - The database instance exists.
  - The database instance is populated with current NNMi data.

Use Oracle tools to copy NNMi data from the working database instance to the target database instance.

  - The database instance is running.

### Task 2: Update the NNMi Configuration

1 Back up the database connection configuration file:

a Change to the following directory:

— *Windows*: %NnmInstallDir%\nonOV\jboss\nms\server\nms\

— *UNIX*: \$NnmInstallDir/nonOV/jboss/nms/server/nms/

b Within the nms directory, create a directory called `deploy.save`.

c Copy the `nms-ds.xml` file from the `deploy` directory to the `deploy.save` directory.



At startup, the `ovjboss` process reads all files in the `deploy` directory hierarchy. For this reason, save backup copies of the deployed files in a location outside of the `deploy` directory hierarchy, as we do here with the `deploy.save` directory.

2 Edit the database connection configuration file:

a Change to the `deploy` directory.

- b In any text editor, open the `nms-ds.xml` file.
- c Locate the `connection-url` entry.

For example:

```
<connection-url>jdbc:oracle:thin:@ohost:1521:nnmidb1</connection-url>
```

The last three parameters in this entry are of interest. They are of the format `oracle_hostname:database_port:database_instance_name`

- d Change one or more of the fourth, fifth, and sixth parameters in the `connection-url` entry.

For example:

- To point to a different Oracle database server, change `ohost` to another hostname.
- To connect to the Oracle database server on a different port, change `1521` to another port number.
- To connect to a different Oracle database instance, change `nnmidb1` to another database instance name. (This database instance must already exist!)

- e Save the `nms-ds.xml` file.

- 3 Start NNMi:

```
ovstart
```

---

## Changing the Password that NNMi Uses to Connect to the Oracle Database Instance

If you change the Oracle configuration to use a different password for connecting to the NNMi database instance, update the NNMi configuration by following these steps:

- 1 Shut down NNMi:

```
ovstop
```

- 2 Run the `nmchangedbpw.ovpl` command and follow the prompts.

- 3 Start NNMi:

```
ovstart
```

For more information, see the `nmchangedbpw.ovpl` reference page, or the UNIX manpage.





# Moving NNMi from Red Hat Linux 4.6 to 5.2 or 5.3

NNMi 9.00 does not support Red Hat Linux 4.6. You must change the operating system to Red Hat Linux 5.2 or 5.3 before migrating to NNMi 9.00.

Use the information in this chapter if you have NNMi 8.1x patch 6 or later running on a Red Hat Linux 4.6 server, and need to change the operating system to Red Hat Linux 5.2 or 5.3.

This chapter contains the following topic:

[Changing NNMi from Red Hat Linux 4.6 to Red Hat Linux 5.2 or 5.3](#)

---

## Changing NNMi from Red Hat Linux 4.6 to Red Hat Linux 5.2 or 5.3

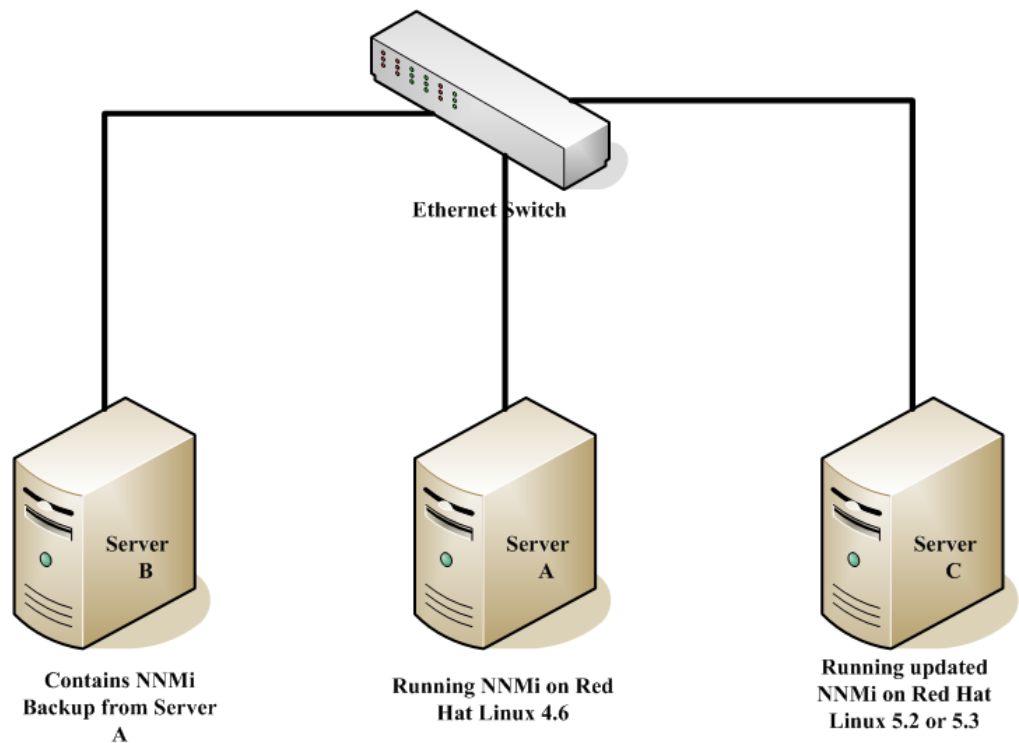
To complete the following steps, you must have NNMi 8.1x patch 6 or later running on a Linux Red Hat 4.6 server. To check the NNMi version number, note the current patch level in the **About Network Node Manager i-series** window. Verify that the version is 8.13.006 or later. If the version is earlier than that, do not proceed. You need to install NNMi 8.1x patch 6 or later before proceeding.

To change an NNMi management server running NNMi 8.1x patch 6 or later from Red Hat Linux 4.6 to Red Hat Linux 5.2 or 5.3, follow these steps:

- 1 Identify three servers that you will use during this procedure:
  - Server A is the current NNMi management server running Linux Red Hat 4.6.
  - Server B will hold the NNMi backup files.
  - Server C will become the new NNMi management server running Linux Red Hat 5.2 or 5.3. This NNMi management server can be the same hardware as the current Server A.

Make sure the `/etc/hosts` file on the new NNMi management server contains the following entry:

```
127.0.0.1 localhost
```



- 2 On Server A, run the `nnmbackup.ovpl -type online -scope all -target /tmp/bak/all` command to complete a full NNMi backup.

For more information about which command options to use, see “NNMi Backup and Restore Tools” in the *NNMi Deployment Reference* and the `nnmbackup.ovpl` reference page, or the UNIX manpage.

- 3 On Server A, copy the backup you completed in [step 2](#) to Server B.
- 4 On Server C, install Red Hat Linux 5.2 or 5.3.

▶ As an alternative to using Server C, reformat the disk on Server A and install Red Hat Linux 5.2 or 5.3. If you do that, substitute Server A for Server C for the remaining steps.

- 5 On Server C, install NNMi 8.10.

See *Installing NNMi 8.10 on Red Hat 5.2* in the *NNMi 8.1x Patch 4 Installation Guide for Linux* for information about completing this step.

- 6 On Server C, install 8.1x patch 6 or later. You must install the same patch level that NNMi Server A was at during the backup you completed in [step 2](#).
- 7 On Server B, copy the NNMi backup to Server C.

- 8 On Server C, run the `nnmrestore.ovpl -force -source /tmp/bak/all` command to complete a full NNMi restore.

For more information about which command options to use, see “NNMi Backup and Restore Tools” in the *NNMi Deployment Reference* and the `nnmrestore.ovpl` reference page, or the UNIX manpage.

▶ Use the command options that match the backup you completed in [step 2](#)

- 9 NNMi associates its license keys with a server's IP address. If the IP address for Server C is different from the IP address of Server A, obtain and install new NNMi license keys. See "Changing the IP Address of a Standalone NNMi Management Server" in the *NNMi Deployment Reference*.



# Migrating NNMi Oracle Data

Suppose you must move the Oracle data used by your NNMi management server from one Oracle database instance to another. One example of this is to move NNMi data from an Oracle 10g database to an Oracle 11g database. The information in this chapter explains the steps to take to complete this work.

---

## Migrating NNMi Oracle Data

Suppose you have NNMi running in one of the following configurations:

- NNMi 8.1x with the latest patch connected to an Oracle 10g database and you must upgrade to NNMi 9.0x.
- NNMi 9.0x connected to an Oracle 10G or Oracle 11G database.

The Oracle database instance migration you must complete could include combinations of the following requirements:

- The existing Oracle instance can be running Oracle 10G or 11G.
- The new Oracle instance can be running Oracle 10G or 11G. You cannot move an existing Oracle 11G instance back to Oracle 10G.
- The new Oracle instance can be located on the original server or on a different server and hostname.

➤ NNMi 8.1x cannot connect to an Oracle 11G server.

To complete the migration of the NNMi Oracle data, complete the following steps:

- 1 As root or administrator, run the following command to stop NNMi: **ovstop -c**.
- 2 Use Oracle tools to move or copy the NNMi data from the existing Oracle server to the new server. Refer to Oracle documentation for additional information.

➤ This Oracle data migration can be an in-place upgrade from Oracle 10 to Oracle 11 on the same server. Oracle provides database migration tools for converting Oracle 10 data into the Oracle 11 format.

- 3 Only complete this step if the new Oracle server has a different hostname than the previous Oracle server. On the NNMi management server, reconfigure NNMi to point to the new Oracle server by completing the following steps:

- a Edit the datasource configuration file shown below:



It is important that you complete the following steps accurately, or jboss will not correctly connect to the Oracle 11G database.

— *Windows*: %NNM\_JBOSS%\server\nms\deploy\nms-ds.xml

— *UNIX*: \$NNM\_JBOSS/server/nms/deploy/nms-ds.xml

- b Change the following attribute to reflect your new server

OLD:

```
<connection-url>jdbc:oracle:thin:@EXISTING_FQDN:EXISTING_ORACLE_PORT:EXISTING_SID </connection-url>
```

NEW:

```
<connection-url>jdbc:oracle:thin:@NEW_FQDN:NEW_PORT:NEW_SID</connection-url>
```

- 4 Complete one of the following actions:

If you are upgrading from NNMi 8.1x to NNMi 9.0x, perform that migration now, following the installation instructions in the *HP Network Node Manager i Software Installation Guide*.

If you are already using NNMi 9.0x, follow these steps to restart NNMi and complete the Oracle database move/migration:

- a Run the following command on the NNMi management server to restart NNMi: **ovstart -c**
- b Run the following command on the NNMi management server to check if all of the services are started and operating correctly: **ovstatus -v**

# Additional Upgrade Information

This chapter describes some changes between NNMi 9.00 and earlier NNMi versions. This chapter contains the following topics:

- [Configuration Differences](#)
- [Functionality Differences](#)

---

## Configuration Differences

After upgrading, you can find many of the configuration files from earlier version of NNMi in new locations.

- After upgrading, you can find most properties files that influence NNMi 9.00 behavior at the following locations:
  - *Windows:* %NNM\_DATA%\shared\nnm\conf\props
  - *Windows:* %NNM\_DATA%\conf\nnm\props
  - *UNIX:* \$NNM\_DATA/shared/nnm/conf/props
  - *UNIX:* \$NNM\_DATA/conf/nnm/props/
- To modify the ovjboss process startup JVM options, such as heap size, edit the following file:
  - *Windows:* %NNM\_DATA%\shared\nnm\conf\props\ovjboss.jvmargs
  - *UNIX:* \$NNM\_DATA%/shared/nnm/conf/props/ovjboss.jvmargs
- To modify trap server properties, edit the following file:
  - *Windows:*  
%NNM\_DATA%\shared\nnm\conf\props\nnmtrapserver.properties
  - *UNIX:* \$NNM\_DATA/shared/nnm/conf/props/nnmtrapserver.properties

- During an upgrade to NNMi 9.00, NNMi preserves the `nms-jboss.properties` file contents. The `ovjboss.jvm.properties` file has a new location:
  - *Windows*: `%NNM_DATA%\shared\nnm\conf\props\nms-jboss.properties`
  - *UNIX*: `$NNM_DATA/shared/nnm/conf/props/nms-jboss.properties`
- To modify application failover properties, edit the following file:
  - *Windows*: `%NNM_DATA%\shared\nnm\conf\props\nms-jboss.properties`
  - *UNIX*: `$NNM_DATA/shared/nnm/conf/props/nms-jboss.properties`
- To modify port properties that were located in the `port.properties` file, edit the following file:
  - *Windows*: `%NNM_DATA%\conf\nnm\props\nms-local.properties`
  - *UNIX*: `$NNM_DATA/conf/nnm/props/nms-local.properties`
- You now select node group status using a check box on the **Node Group** configuration form. After you upgrade an NNMi management server, NNMi retains the existing node groups the way they were before the upgrade.

---

## Functionality Differences

- Many commands and scripts now require a username and password to run. For more information, see the reference page or the UNIX manpage for the command or script you want to run.
- NNMi does not start the `nmsdbmgr` process if it is using an Oracle database.
- Dampening settings are no longer disabled out-of-the-box.
  - Dampening is turned on for most management events.
- You can use the `nnmsetdampenedinterval.ovpl` script to adjust the dampening interval. This script sets the dampening interval for all management event configurations. See the `nnmsetdampenedinterval.ovpl` reference page or the UNIX manpage for more information.
  - After upgrading, the `nnmsetdampenedinterval.ovpl` script is most useful for any of the integrations that use the NNMi northbound interface:
    - NNMi northbound interface
    - NNMi Integration Module for Netcool Software
    - HPOM agent implementation of the HP NNMi-HPOM integration

For dampening, write down the value of the **Holding Period** parameter for the integration configuration before installing NNMi 9.00. After upgrading, run the `nnmsetdampenedinterval.ovpl` script to apply this value across NNMi.

- If you upgrade to NNMi 9.00, and have a different dampening period set (something other than 6 minutes), you can globally reset all dampened intervals to a different value using the `nnmsetdampenedinterval.ovpl` script.

This is a manual step. It does not happen automatically during an upgrade.



- NNMi 9.00 does not include the NodeUp management event incident. The upgrade to NNMi 9.00 retains the incident configuration, but the NNMi root cause analysis no longer triggers the NodeUp incident.
  - If you need notification that a node is up, associate a lifecycle transition action with the CLOSED lifecycle state of the NodeDown incident. In most cases, you can transfer the action for the NodeUp incident REGISTERED state to the NodeDown incident CLOSED state with little or no change.
  - Integrations that use the NNMi northbound interface (including the NNMi Integration Module for Netcool Software), can receive traps that indicate when a NodeDown incident has been closed.
- NNMi 9.00 adds the **Calculate Status** setting to the **Node Group** configuration form. Upgrading to NNMi 9.00 selects the **Calculate Status** check box for all existing node groups.
  - Consider disabling the **Calculate Status** setting for large node groups, particularly the **Network Infrastructure Devices** node group, as node group status calculation can be expensive resource-wise for large environments.
  - See *Check Status Details for a Node Group* in the NNMi help for information about checking status for a node group.
- After upgrading to NNMi 9.00, NNMi uses ICMP (ping) of management addresses.
- You can configure State Poller data collection to be based on an ICMP (ping) response, or to be based on SNMP data.
- Device profile configuration upgrades from NNMi 8.x can modify some settings. If you do not want these values modified during an upgrade, change the **Author** field to some value different from HP Network Node Manager.
- URL action configuration upgrades from NNMi 8.x can modify some settings. If you do not want these values to be modified during an upgrade, change the **Author** field to some value different from HP Network Node Manager.
- NNMi 9.00 adds a configuration form for the HPOM agent implementation of the HP NNMi-HPOM integration. For long-term maintenance purposes, it is recommended that you transfer the integration configuration from the **HP NNMi–Northbound Interface Destination** form to the **HP NNMi–HPOM Agent Destination** form. After transferring the configuration, delete the destination from the **HP NNMi–Northbound Interface Destination** form.
- Most processes now log messages to the `nm.0.0.log` file, instead of to separate log files for each component. For more information, see “NNMi Logging” in the *NNMi Deployment Reference*.



# Application Failover and Upgrading from NNMi 8.x to NNMi 9.0x

## Application Failover and Upgrading to NNMi 9.00

If you plan to upgrade an earlier version of NNMi 8.1x that is running in an NNMi application failover configuration, the supported upgrade path is to temporarily unconfigure application failover, upgrade the NNMi management server to NNMi 9.00, then reconfigure application failover.

To upgrade NNMi management servers configured for application failover, follow these steps:

- 1 As a precaution, run the **nnmconfigexport.ovpl** script on both the active and standby NNMi management servers before proceeding. For information, see “Best Practice: Save the Existing Configuration” in the *NNMi Deployment Reference*.
- 2 As a precaution, back up your NNMi data on both the active and standby NNMi management servers before proceeding. For information, see “Backup Scope” in the *NNMi Deployment Reference*.
- 3 As a precaution, on the active NNMi management server, complete the following steps:
  - a Run the **nnmcluster** command.
  - b Embedded database only: After NNMi prompts you, type **dbsync**, then press Enter. Review the displayed information to make sure it includes the following messages:

ACTIVE\_DB\_BACKUP: This means that the active NNMi management server is performing a new backup.

ACTIVE\_NNM\_RUNNING: This means that the active NNMi management server completed the backup referred to by the previous message.

STANDBY\_READY: This shows the previous status of the standby NNMi management server.

STANDBY\_RECV\_DBZIP: This means that the standby NNMi management server is receiving a new backup from the active NNMi management server.

STANDBY\_READY: This means that the standby NNMi management server is ready to perform if the active NNMi management server fails.

- 4 Run the **nnmcluster -halt** command on the active NNMi management server. This shuts down all `nnmcluster` processes on both the active and standby NNMi management servers.
- 5 To verify there are no `nnmcluster` nodes running on either server, *complete the following steps on both the active and standby NNMi management servers.*
  - a Run the **nnmcluster** command.
  - b Verify that there are no `nnmcluster` nodes present except the one marked (SELF).
  - c Run **exit** or **quit** to stop the interactive `nnmcluster` process you started in step a.
- 6 *Complete the following steps on both the active and standby NNMi management servers to disable application failover:*
  - a Edit the following file:
    - *Windows:* %NNM\_SHARED\_CONF%\ov.conf
    - *UNIX:* \$NNM\_SHARED\_CONF/ov.conf
  - b Comment out the `com.hp.ov.nms.cluster.name` parameter.
  - c Write down the value of the `com.hp.ov.nms.cluster.name` parameter. You need that value in a later step.
  - d Save your changes.
- 7 Upgrade the active NNMi management server using the instructions located in the *NNMi Installation Guide*.
- 8 Run the **ovstart** command on the active NNMi management server.
- 9 Upgrade the standby NNMi management server by following the instructions in the *NNMi Installation Guide*.
- 10 Run the **ovstart** command on the standby NNMi management server.
- 11 *Complete the following steps on both the active and standby NNMi management servers:*
  - a Run the **ovstop** command.
  - b Edit the following file:
    - *Windows:* %NNM\_SHARED\_CONF%\props\nms-cluster.properties
    - *UNIX:* \$NNM\_SHARED\_CONF/props/nms-cluster.properties
  - c Type in the value of the `com.hp.ov.nms.cluster.name` parameter you wrote down in [step c](#) on page 100.
  - d Uncomment the `com.hp.ov.nms.cluster.name` parameter.
  - e Save your changes.
- 12 Run the **ovstart** command on the active NNMi management server only. To verify that this step is complete, run the **nnmcluster -display** command on the active NNMi management server and look for an `ACTIVE_NNM_RUNNING` message.
- 13 After you complete [step 12](#) on the active NNMi management server, run the **ovstart** command on the standby NNMi management server to finish enabling application failover.

- 14 If you installed the NNM iSPI Performance for QA, the NNM iSPI Performance for Metrics, or the NNM iSPI Performance for Traffic; are using the application failover feature; and completed the upgrade process shown above, run the NNM iSPI enablement script for each NNM iSPI on both the active and standby NNMi management servers.
- 15 If you are using Linux NNMi management servers, run the following command on both the active and standby NNMi management servers:  
`chmod 777 /var/opt/OV/shared/perfSpi/datafiles/nnm_details.xml`

## Application Failover and NNMi Patches

To apply patches to the NNMi management servers configured for application failover, follow these steps:

- 1 As a precaution, run the `nnmconfigexport.ovpl` script on both the active and standby NNMi management servers before proceeding. For information, see “Best Practice: Save the Existing Configuration” in the *NNMi Deployment Reference*.
- 2 As a precaution, back up your NNMi data on both the active and standby NNMi management servers before proceeding. For information, see “Backup Scope” in the *NNMi Deployment Reference*.
- 3 As a precaution, on the active NNMi management server, do the following steps:
  - a Run the `nnmcluster` command.
  - b Embedded database only: After NNMi prompts you, type `dbsync`, then press Enter. Review the displayed information to make sure it includes the following messages:  
 ACTIVE\_DB\_BACKUP: This means that the active NNMi management server is performing a new backup.  
 ACTIVE\_NNM\_RUNNING: This means that the active NNMi management server completed the backup referred to by the previous message.  
 STANDBY\_READY: This shows the previous status of the standby NNMi management server.  
 STANDBY\_RECV\_DBZIP: This means that the standby NNMi management server is receiving a new backup from the active NNMi management server.  
 STANDBY\_READY: This means that the standby NNMi management server is ready to perform if the active NNMi management server fails.
- 4 Run the `nnmcluster -halt` command on the active NNMi management server. This shuts down all `nnmcluster` processes on both the active and standby NNMi management servers.
- 5 To verify there are no `nnmcluster` nodes running on either server, *complete the following steps on both the active and standby NNMi management servers.*
  - a Run the `nnmcluster` command.
  - b Verify that there are no `nnmcluster` nodes present except the one marked (SELF).
  - c Run `exit` or `quit` to stop the interactive `nnmcluster` process you started in step a.
- 6 On the active NNMi management server, comment out the `com.hp.ov.nms.cluster.name` parameter in the `nms-cluster.properties` file.
  - a Edit the following file:

- *Windows*: %NNM\_SHARED\_CONF%\props\nms-cluster.properties
  - *UNIX*: \$NNM\_SHARED\_CONF/props/nms-cluster.properties
- b Comment out the `com.hp.ov.nms.cluster.name` parameter.
  - c Save your changes.
- 7 Apply the NNMi patch to the active NNMi management server using the instructions provided with the patch.
  - 8 On the active NNMi management server, uncomment the `com.hp.ov.nms.cluster.name` parameter in the `nms-cluster.properties` file.
    - a Edit the following file:
      - *Windows*: %NNM\_SHARED\_CONF%\props\nms-cluster.properties
      - *UNIX*: \$NNM\_SHARED\_CONF/props/nms-cluster.properties
    - b Uncomment the `com.hp.ov.nms.cluster.name` parameter.
    - c Save your changes.
  - 9 Run the `ovstart` command on the active NNMi management server.
  - 10 Verify that the patch installed correctly on the active NNMi management server by viewing information on the **Product** tab of the **Help > System Information** window in the NNMi console.
  - 11 Run the `nnmcluster -dbsync` command to create a new backup.
  - 12 On the standby NNMi management server, comment out the `com.hp.ov.nms.cluster.name` parameter in the `nms-cluster.properties` file as shown in [step a](#) on page 101 through [step c](#) on page 102.
  - 13 Apply the NNMi patch to the standby NNMi management server.
  - 14 On the standby NNMi management server, uncomment the `com.hp.ov.nms.cluster.name` parameter in the `nms-cluster.properties` file as shown in [step a](#) on page 102 through [step c](#) on page 102.
  - 15 Run the `ovstart` command on the standby NNMi management server.
  - 16 If you installed the NNM iSPI Performance for QA, the NNM iSPI Performance for Metrics, or the NNM iSPI Performance for Traffic; are using the application failover feature; and completed the patch process shown above, run the NNM iSPI enablement script for each NNM iSPI on both the active and standby NNMi management servers.
  - 17 If you are using Linux NNMi management servers, run the following command on both the active and standby NNMi management servers:
 

```
chmod 777 /var/opt/OV/shared/perfSpi/datafiles/nnm_details.xml
```

# High Availability and Upgrading from NNMi 8.1x to NNMi 9.0x

---

## Upgrading NNMi under HA from NNMi 8.1x to NNMi 9.01



This procedure references the NNMi 8.1x version of this document for making changes to the NNMi management server before upgrading NNMi. The *NNMi Deployment Guide* for NNMi 8.1x is available from <http://h20230.www2.hp.com/selfsolve/manuals>. See [Available Product Documentation](#) on page 3 for more information.

To upgrade from NNMi 8.1x under HA to NNMi 9.01 under HA, upgrade the active node, fail over from the active node to the passive node, and then upgrade the second node. Follow these steps:

- 1 Ensure that the fully-qualified domain name is correctly set on each NNMi management server in the NNMi HA resource group. On each NNMi management server, run the following command:

```
nnmofficialfqdn.ovpl -t
```

- If the returned value is the virtual hostname of the NNMi HA resource group, continue with [step 2](#) of this procedure.
- If the returned value is not the virtual hostname of the NNMi HA resource group, update the configuration of each NNMi management server in the HA cluster as described in the *NNMi Deployment Guide* for NNMi 8.1x. (See “Changing the Hostname or Domain Name of an NNMi Management Server” in the “Changing the NNMi Management Server” chapter.)



In the referenced procedure, do not rename or reboot the NNMi management server.

- 2 Use the `nnmbackup.ovpl` command, or another database command, to back up all NNMi data. For example:

```
nnmbackup.ovpl -type offline -scope all -target nmi_backups
```

For more information about this command, see the *nnmbackup.ovpl* reference page, or the UNIX manpage.

- 3 Ensure that the NNMi 8.1x configuration is consistent across all HA nodes by forcing a failover, in turn, to each of the passive nodes.
- 4 Ensure that all nodes in the NNMi 8.1x HA cluster are running NNMi 8.1x Patch 8 or a higher version of NNMi 8.1x.

If necessary, upgrade each system to the latest NNMi 8.1x consolidated patch. Follow the instructions in the “Patching NNMi under HA” section of the “Configuring NNM i-series Software in a High Availability Cluster” chapter in the most recent NNMi 8.1x version of the *NNMi Deployment Guide*.

- 5 Determine which node in the NNMi 8.1x HA resource group is active:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \
-group <resource_group> -activeNode
```


- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \
-group <resource_group> -activeNode
```

The remainder of this procedure refers to the currently active node (the node identified by the `nmhaclusterinfo.ovpl` command) as server X and the currently passive node as server Y.

- 6 On server X (which is the original active node), disable NNMi HA resource group monitoring by creating the following maintenance file:

- *Windows:* %NnmDataDir%\hacluster\*<resource\_group>*\maintenance
- *UNIX:* \$NnmDataDir/hacluster/*<resource\_group>*/maintenance

 The first line of the maintenance file must contain only the single word: NORESTART

- 7 On server X, upgrade NNMi:

- a Stop NNMi:

```
ovstop -c
```

- b Install NNMi 9.00 as described in [Upgrading from NNMi 8.0x or 8.1x](#) on page 71.

The NNMi database on the shared disk is upgraded to the format of the new NNMi product version at this time.

- c Apply the latest consolidated NNMi patch as described in the patch installation instructions.
- d Upgrade all add-on NNM iSPIs to version 9.00 as described in the installation guide or the deployment guide for each NNM iSPI.
- e Apply the latest consolidated patch for each installed NNM iSPI as described in the patch installation instructions.



If your environment includes standalone NNM iSPIs, you must also upgrade those products to version 9.00 for correct functionality. You can complete those upgrades after completing this procedure.



- 8 On server Y (which is still operating as the passive node), disable HA resource group monitoring by creating the following maintenance file:

- *Windows:* %NnmDataDir%\hacluster\*<resource\_group>*\maintenance
- *UNIX:* \$NnmDataDir/hacluster/*<resource\_group>*/maintenance



The first line of the maintenance file must contain only the single word:  
NORESTART

- 9 Move control of the NNMi HA resource group to server Y:

- *MSFC* or *MSCS*:
  - On server X, take the NNMi HA resource group offline.
  - On server X, move the NNMi HA resource group to server Y.
  - On server Y, bring online all resources *except* the resource group application.
- *Serviceguard*:
  - On server X, force a failover to server Y.
- *VCS*:
  - On server X, take the NNMi HA resource group offline.
  - On server X, move the NNMi HA resource group to server Y.
  - On server Y, bring online all resources *except* the resource group application.

- 10 On server Y (which is now the active node), upgrade NNMi:

- a Install NNMi 9.00 as described in [Upgrading from NNMi 8.0x or 8.1x](#) on page 71.
  - b Apply the latest consolidated NNMi patch as described in the patch installation instructions.
  - c Upgrade all add-on NNM iSPIs to version 9.00 as described in the installation guide or the deployment guide for each NNM iSPI.
  - d Apply the latest consolidated patch for each installed NNM iSPI as described in the patch installation instructions.
- 11 If the HA cluster includes multiple passive nodes, repeat [step 8](#) through [step 10](#) for each passive node.

- 12 *Optional.* Force a failover from server Y to server X so that the node that was active before the upgrade process is again the active node.

- 13 Start NNMi:

```
ovstart
```

- 14 Verify that NNMi started correctly:

```
ovstatus -c
```

All NNMi services should show the state RUNNING.

- 15 On all servers, delete the maintenance file:

- *Windows:* %NnmDataDir%\hacluster\*<resource\_group>*\maintenance
- *UNIX:* \$NnmDataDir/hacluster/*<resource\_group>*/maintenance

---

# Unconfiguring NNMi from an HA Cluster

## Unconfiguring NNMi from an HA Cluster

The process of removing an NNMi node from an HA cluster involves undoing the HA configuration for that instance of NNMi. You can then run that instance of NNMi as a standalone management server, or you can uninstall NNMi from that node.

If you want to keep NNMi configured for high availability, the HA cluster must contain one node that is actively running NNMi and at least one passive NNMi node. If you want to completely remove NNMi from the HA cluster, unconfigure the HA functionality on all nodes in the cluster.

To completely unconfigure NNMi from an HA cluster, follow these steps:

- 1 Determine which node in the HA cluster is active. On any node, run the following command:
  - *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \  
-group <resource_group> -activeNode
```
  - *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \  
-group <resource_group> -activeNode
```
- 2 On each passive node, unconfigure any add-on NNM iSPIs from the HA cluster. For information, see the documentation for each NNM iSPI.

- 3 On any node in the HA cluster, verify that the add-on NNM iSPIs on all passive nodes have been unconfigured from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \
-config NNM -get NNM_ADD_ON_PRODUCTS
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \
-config NNM -get NNM_ADD_ON_PRODUCTS
```

The command output lists the add-on iSPI configurations in the format `<iSPI_PM_Name>[hostname_list]`. For example:

```
PerfSPIHA[hostname1, hostname2]
```

At this time, only the active node hostname should appear in the output. If a passive node hostname appears in the output, repeat [step 2](#) until this command output includes only the active node hostname.

- 4 On each passive node, unconfigure NNMi from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM \
<resource_group>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM \
<resource_group>
```

This command removes access to the shared disk but does not unconfigure the disk group or the volume group.

- 5 On each passive node, move the NNMi HA resource group-specific files to a separate location for safe-keeping:



If you do not plan to reconfigure the NNMi HA resource group, you do not need to save a copy of these files, and you can delete them at this time.

- *MSFC or MSCS:* In Windows Explorer, delete the `%NnmDataDir%\hacluster\<resource_group>` folder.

- *Serviceguard:*

— *HP-UX:*

```
rm -rf /var/opt/OV/hacluster/<resource_group>
rm -rf /etc/cmcluster/<resource_group>
```

— *Linux:*

```
rm -rf /var/opt/OV/hacluster/<resource_group>
rm -rf /usr/local/cmcluster/conf/<resource_group>
```

- *VCS:*

```
rm -rf /var/opt/OV/hacluster/<resource_group>
```

- 6 On the active node, unconfigure any add-on NNM iSPIs from the HA cluster.

For information, see the documentation for each NNM iSPI. On any node in the HA cluster, verify that the add-on NNM iSPIs on all nodes have been unconfigured from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \
-config NNM -get NNM_ADD_ON_PRODUCTS
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \
-config NNM -get NNM_ADD_ON_PRODUCTS
```

If any hostname appears in the output, repeat [step 6](#) until this command output indicates that no iSPIs are configured.

- 7 On the active node, stop the NNMi HA resource group:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhastoprg.ovpl NNM \
<resource_group>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhastoprg.ovpl NNM \
<resource_group>
```

This command does not remove access to the shared disk. Nor does it unconfigure the disk group or the volume group.

- 8 On the active node, unconfigure NNMi from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM \
<resource_group>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM \
<resource_group>
```

This command removes access to the shared disk but does not unconfigure the disk group or the volume group.

- 9 On the active node, move the NNMi HA resource group-specific files to a separate location for safe-keeping:



If you do not plan to reconfigure the NNMi HA resource group, you do not need to save a copy of these files, and you can delete them at this time.

- *MSFC or MSCS:* In Windows Explorer, delete the %NnmDataDir%\hacluster\*<resource\_group>* folder.

- *Serviceguard:*

— *HP-UX:*

```
rm -rf /var/opt/OV/hacluster/ <resource_group>
rm -rf /etc/cmcluster/ <resource_group>
```

— *Linux:*

```
rm -rf /var/opt/OV/hacluster/<resource_group>
rm -rf /usr/local/cmcluster/conf/<resource_group>
```

- *VCS:*

```
rm -rf /var/opt/OV/hacluster/<resource_group>
```

#### 10 Unmount the shared disk.

- If you want to reconfigure the NNMi HA cluster at some point, you can keep the disk in its current state.
- If you want to use the shared disk for another purpose, copy all data that you want to keep (as described in [Running NNMi with the Existing Database Outside HA](#) on page 109), and then use the HA product commands to unconfigure the disk group and volume group.

## Running NNMi with the Existing Database Outside HA

If you want to run NNMi outside HA on any node with the existing database, follow these steps:

- 1 On the active node (if one still exists), ensure that NNMi is not running:

```
ovstop
```

Alternatively, check the status of the `ovspmd` process by using Task Manager (Windows) or the `ps` command (UNIX).

- 2 On the current node (where you want to run NNMi outside HA), verify that NNMi is not running:

```
ovstop
```



To prevent data corruption, make sure that no instance of NNMi is running and accessing the shared disk.

- 3 (UNIX only) Activate the disk group:

```
vgchange -a e <disk_group>
```

- 4 Use the appropriate operating system commands to mount the shared disk. For example:

- *Windows:* Use Windows Explorer.
- *UNIX:* `mount /dev/vgnm/lvnm /nnmmount`

- 5 Copy the NNMi files from the shared disk to the node:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM \
-from <HA_mount_point>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM \
-from <HA_mount_point>
```

- 6 Use the appropriate operating system commands to unmount the shared disk. For example:
  - *Windows*: Use Windows Explorer.
  - *UNIX*: `umount /nmmount`
- 7 (UNIX only) Deactivate the disk group:

```
vgchange -a n <disk_group>
```

- 8 Obtain and install the permanent production license keys for the physical IP address of this NNMi management server.
- 9 Start NNMi:

```
ovstart -c
```

NNMi is now running with a copy of the database that was formerly used by the NNMi HA resource group. Manually remove from the NNMi configuration any nodes that you do not want to manage from this NNMi management server.

---

## Patching NNMi under HA

To apply a patch for NNMi, work in HA maintenance mode. Follow these steps:

- 1 Determine which node in the HA cluster is active:
  - *Windows*:
 

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \  
-group <resource_group> -activeNode
```
  - *UNIX*:
 

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \  
-group <resource_group> -activeNode
```
- 2 On the active node, put the NNMi HA resource group into maintenance mode as described in “Putting an HA Resource Group into Maintenance Mode” in the *NNMi Deployment Reference*.
 

Include the `NORESTART` keyword.
- 3 On all passive nodes, put the NNMi HA resource group into maintenance mode as described in “Putting an HA Resource Group into Maintenance Mode” in the *NNMi Deployment Reference*.
 

Include the `NORESTART` keyword.
- 4 On the active node, follow these steps:
  - a Stop NNMi:
 

```
ovstop -c
```
  - b Back up the shared disk by performing a disk copy.
  - c *Optional*. Use the `nnmbackup.ovpl` command, or another database command, to back up all NNMi data. For example:
 

```
nnmbackup.ovpl -type offline -scope all -target nnm_backups
```

For more information about this command, see “NNMi Backup and Restore Tools” in the *NNMi Deployment Reference*.

- d Apply the appropriate NNMi and NNM iSPI patches to the system.
- e Start NNMi:

**ovstart -c**

- f Verify that NNMi started correctly:

**ovstatus -c**

All NNMi services should show the state `RUNNING`.

- 5 On each passive node, apply the appropriate patches to the system.



Never run the `ovstart` or `ovstop` commands on a secondary (backup) cluster node.

- 6 On all passive nodes, take the NNMi HA resource group out of maintenance mode as described in “Removing an HA Resource Group from Maintenance Mode” in the *NNMi Deployment Reference*.
- 7 On the active node, take the NNMi HA resource group out of maintenance mode as described in “Removing an HA Resource Group from Maintenance Mode” in the *NNMi Deployment Reference*.





# Upgrading from NNMi 9.0x or 9.1x

For information about upgrading from NNM 6.x or 7.x to NNMi 9.20, see [Upgrading from 6.x or 7.x](#) on page 13.

For information about upgrading from NNMi 8.1x to NNMi 9.20, see [Upgrading from NNMi 8.0x or 8.1x](#) on page 71.

You can upgrade NNMi according to the information shown in [Table 8](#). The information shown in [Table 8](#) assumes you have NNMi 9.0x Patch 5 or newer installed on the NNMi management server.

**Table 8 Supported NNMi Upgrades**

NNMi Version	Upgrade to NNMi 9.20
NNMi 9.0x Patch 5 or newer	Supported
NNMi 9.1x Patch 3 or newer	Supported

To upgrade from NNMi 9.0x or 9.1x to NNMi 9.20, you must upgrade directly to NNMi 9.20. During an upgrade from NNMi 9.0x or 9.1x to NNMi 9.20, the installation script provides an opportunity to install patches for NNMi 9.20, as applicable.

If you are upgrading from NNMi 9.0x or 9.1x and you also have the Master or Leaf Collector for the NNM iSPI Performance for Traffic installed on the NNMi management server, you must upgrade to NNMi 9.10 patch 3 (or later) and NNM iSPI Performance for Traffic 9.10 patch 2 (or later) before upgrading to NNMi 9.20. Failure to do so will result in the loss of all traffic data.

If you plan to upgrade an earlier version of NNMi that is running in an NNMi application failover configuration, the supported upgrade path is to temporarily unconfigure application failover, upgrade the NNMi management server to NNMi 9.20, and then reconfigure application failover. For detailed information, see [Application Failover and Upgrading to NNMi 9.20](#) on page 137.

If you added `com.sun.management.jmxremote.*` properties to any of the properties files in the following directories, NNMi does not retain these values during an upgrade to NNMi 9.20:

- *Windows*: %NNM\_DATA%\shared\nnm\conf\props
- *UNIX*: \$NNM\_DATA/shared/nnm/conf/props

NNMi 9.20 ignores any `com.sun.management.jmxremote.*` properties you add to the properties files in these directories.

If you plan to upgrade an earlier version of NNMi that is running under high availability (HA), see [High Availability and Upgrading from NNMi 9.0x or 9.1x to NNMi 9.20](#) on page 147.

If you plan to upgrade NNMi management servers configured in a global network management environment see [Upgrading Global and Regional Managers from NNMi 9.0x or 9.1x](#) on page 135.

If you plan to upgrade a Linux NNMi management server from NNMi 9.0x or 9.1x to NNMi 9.20, you must import the HP public key into the Linux RPM database before installing NNMi 9.20. To do this, point your browser to the following location and follow the instructions:

<https://h20392.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=HPLinuxCodeSigning>



If you created a User Group in NNMi 9.1x called `globalops`, then that group, in NNMi 9.20, will now have access to all topology objects. If that behavior is not what you desire, then you should rename that User Group before upgrading to NNMi 9.20.

If you have NNMi 9.0x integrated with NA 9.00 and plan to upgrade NNMi from NNMi 9.0x to NNMi 9.10, you must disable the NNMi-NA integration and uninstall the NNMi connector before upgrading. To do this, follow the instructions shown in “Integration Configuration Upgraded from NNMi 9.0x” in the *NNMi—Network Automation Integration Guide*.



Note the following:

- Updates to Status and Incidents could be delayed as NNMi resynchronizes following an upgrade.
- If you see the following message during this resynchronization, it does not indicate a problem:  

The Causal Engine’s large queue size is causing delayed updates to Status and Incidents. This could be due to resynchronization following an upgrade, application failover, restore from backup, or a manual resynchronization.
- Do not stop NNMi during this resynchronization. To ensure resynchronization has completed, keep NNMi running for several hours following the upgrade.

There are several upgrade scenarios you could encounter. This section contains the following chapters:

- [Upgrading the NNMi Management Server in Place](#) on page 117, which describes the following upgrade scenario:
  - Upgrading from NNMi 9.1x to NNMi 9.20 on the same hardware and operating system.
- [Upgrading to a Different NNMi Management Server](#) on page 119, which describes the following upgrade scenario:
  - Upgrading from NNMi 9.1x to NNMi 9.20 on the same version operating system.
- [Moving NNMi from Windows 2003 to Windows 2008](#). NNMi 9.20 does not support Windows 2003. You must change the operating system to Windows 2008 before upgrading to NNMi 9.20.
- [Migrating NNMi Oracle Data](#). Explains the steps to take to move the Oracle data used by your NNMi management server from one Oracle database instance to another.
- [Upgrading Global and Regional Managers from NNMi 9.0x or 9.1x](#). Explains the requirements for upgrading in Global Network Management environments.

- [High Availability and Upgrading from NNMi 9.0x or 9.1x to NNMi 9.20](#). Explains the requirements for upgrading in High Availability environments.
- [Application Failover and Upgrading to NNMi 9.20](#) on page 137. Explains the requirements for upgrading in Application Failover environments.
- [Additional Upgrade Information](#). Explains some areas that NNMi 9.20 differs from earlier versions of NNMi.



# Upgrading the NNMi Management Server in Place

This chapter describes the process for upgrading an existing NNMi management server to NNMi 9.20.

This chapter contains the following topic:

- [Upgrade an Existing NNMi Management Server to NNMi 9.20](#)

---

## Upgrade an Existing NNMi Management Server to NNMi 9.20

Read the NNMi 9.20 *Preinstallation Checklist* chapter in the *NNMi Installation Guide* and [Additional Upgrade Information](#) on page 131 before continuing. There are notable changes to the *NNMi Installation Guide*. For example, if you use an Oracle database instance instead of the embedded database, you should set the FLASHBACK ANY TABLE permission, as this enables NNMi to create restore points during migration.

Read the *HP Network Node Manager i Software System and Device Support Matrix* for the NNMi software you are upgrading to before continuing. You can obtain a copy of this document at <http://h20230.www2.hp.com/selfsolve/manuals>. You must have an HP Passport User ID to access this web site.

The following steps explain how to upgrade an NNMi management server to NNMi 9.20. The following steps assume you have NNMi 9.0 patch 5 or NNMi 9.1 patch 3, or later, running on the NNMi management server.

- 1 Back up the NNMi management server using the `nnmbackup.ovpl` script. Do this as a precaution, as you would only use this backup in the unlikely event of a failed migration. For more information, see the *nnmbackup.ovpl* reference page, or the UNIX manpage.
- 2 *Oracle Database Only:* If the NNMi management server uses an Oracle database, have your Oracle database administrator back up the NNMi data. As mentioned earlier, have your Oracle database administrator set the FLASHBACK ANY TABLE permission, as this enables NNMi to create restore points during migration

- 3 *Oracle Database Only:* Use the `nmconfigexport.ovpl` script to back up configuration information from the NNMi management server. Do this as a precaution, as you would only use this backup in the unlikely event of a failed migration. For more information, see the `nmconfigexport.ovpl` or `nmconfigimport.ovpl` reference pages, or the UNIX manpages.



Never edit a file exported with the `nmconfigexport.ovpl` script before using the `nmconfigimport.ovpl` script to import the file.

- 4 Install NNMi 9.20 on the NNMi management server using instructions from the *NNMi Installation Guide*.



*Oracle Database Only:* If your Oracle database administrator does not set the FLASHBACK ANY TABLE permission, you will see a warning about that missing permission after the install completes. You can ignore this warning.

- 5 Verify that the information from the NNMi management server migrated successfully.

# Upgrading to a Different NNMi Management Server

This chapter describes the process for upgrading to NNMi 9.20 on a new system while maintaining the configuration of the existing NNMi management server.

This chapter contains the following topic:

- [Upgrade to a Different NNMi Management Server](#)

---

## Upgrade to a Different NNMi Management Server

Read the NNMi 9.20 *Preinstallation Checklist* chapter in the *NNMi Installation Guide* and [Additional Upgrade Information](#) on page 131 before continuing. There are notable changes to the *NNMi Installation Guide*. For example, if you use an Oracle database instance instead of the embedded database, you should set the FLASHBACK ANY TABLE permission, as this enables NNMi to create restore points during migration.

The following steps explain how to copy data from an existing NNMi management server to a target NNMi management server. The following steps assume you have NNMi 9.0 patch 5 or NNMi 9.1 patch 3, or later, running on the existing NNMi management server.



If you want to change the Oracle database server, complete that process before or after the upgrade to NNMi 9.20. For information, see [Migrating NNMi Oracle Data](#) on page 129.

- 1 As a precaution, back up the existing (source) NNMi 9.0x or 9.1x management server using the `nnmbackup.ovpl` script. Label this backup for NNMi 9.0x or 9.1x. For more information, see the `nnmbackup.ovpl` reference page, or the UNIX manpage for NNMi 9.0x or 9.1x.
- 2 If the existing (source) NNMi management server uses an Oracle database, have your Oracle database administrator back up the NNMi 9.0x or 9.1x data. As mentioned earlier, have your Oracle database administrator set the FLASHBACK ANY TABLE permission, as this enables NNMi to create restore points during migration.

- 3 Install NNMi 9.20 and the latest consolidated patch (if any) on the source NNMi management server using instructions from the *NNMi Installation Guide*.



*Oracle Database Only:* If your Oracle database administrator does not set the FLASHBACK ANY TABLE permission, you will see a warning about that missing permission after the install completes. You can ignore this warning.

- 4 Verify that NNMi 9.20 is working correctly on the source NNMi management server.
- 5 Back up NNMi 9.20 on the source NNMi management server using the `nmbackup.ovpl` script. Label this backup for NNMi 9.20. You will need it to copy data to the target NNMi management server. For more information, see the *nmbackup.ovpl* reference page, or the UNIX manpage for NNMi 9.20.
- 6 Install NNMi 9.20 and the latest consolidated patch (if any) on the target NNMi management server using instructions from the *NNMi Installation Guide*. To migrate the data from [step 5](#), the target NNMi management server must be running the same operating system version. NNMi does not support data migration to an NNMi management server running on a different operating system.
- 7 Use the `nmrestore.ovpl` script to copy NNMi database information to the target server. For more information, see the *nmrestore.ovpl* reference page, or the UNIX manpage.
- 8 Obtain and install a new license on the target NNMi management server.  
For information, see “Licensing NNMi” in the *NNMi Deployment Reference*.
- 9 Verify that the information from the target NNMi management server migrated successfully from the existing NNMi management server.



# Moving NNMi from Windows 2003 to Windows 2008

NNMi 9.10 and NNMi 9.20 do not support Windows 2003. You must change the operating system to Windows 2008 or Windows 2008 R2 before migrating to NNMi 9.20.

Use the information in this chapter if you have NNMi 9.0x (latest patch) running on a Windows 2003 server, and need to change the operating system to Windows 2008.

This chapter contains the following topic:

[Changing NNMi from Windows 2003 to Windows 2008](#)

---

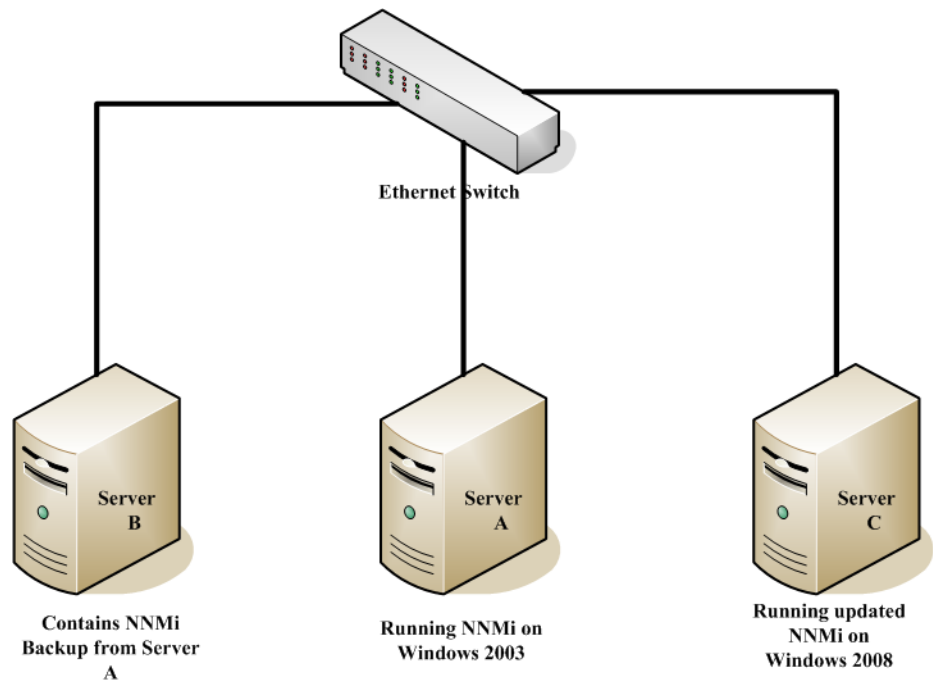
## Changing NNMi from Windows 2003 to Windows 2008

To complete the following steps, you must have NNMi 9.0x (latest patch) running on a Windows 2003 server. To check the NNMi version number, note the current patch level in the **Help->About HP Network Node Manager i Software** window. Verify that the version is 9.01.005 or later. If the version is earlier than that, do not proceed. You need to install NNMi 9.0x (latest patch) before proceeding.

To change an NNMi management server running NNMi 9.0x (patch 5 or later) from Windows 2003 to Windows 2008, follow these steps:

- 1 Identify three servers that you will use during this procedure:
  - Server A is the current NNMi management server running Windows 2003.
  - Server B will hold the NNMi backup files.
  - Server C will become the new NNMi management server running Windows 2008. This NNMi management server can be the same hardware as the current Server A.

Make sure the `hosts` file on the new NNMi management server contains the following entry: **127.0.0.1 localhost**



- 2 On Server A, run the `nnmbackup.ovpl -type online -scope all -target temporary_location` command to complete a full NNMi backup.

For more information about which command options to use, see “NNMi Backup and Restore Tools” in the *NNMi Deployment Reference* and the `nnmbackup.ovpl` reference page, or the UNIX manpage.

- 3 On Server A, copy the backup you completed in [step 2](#) to Server B.
- 4 On Server C, install Windows 2008.



As an alternative to using Server C, reformat the disk on Server A and install Windows 2008. If you do that, substitute Server A for Server C for the remaining steps.

- 5 On Server C, install NNMi 9.0x patch 5 or later. You must install the same patch level that NNMi Server A was at during the backup you completed in [step 2](#).
- 6 During the NNMi installation on server C, the installation script might assign ports that differ from the server B configuration. During the configuration restore on Server C, this might create port conflicts. To remedy this, do the following:
  - a On Server C, navigate to the following directory: `;%$NNM_CONF%\nmm\props\`
  - b On Server C, copy the `nms-local.properties` file to `nms-local.properties.save` in a temporary location.
  - c On Server B, copy the NNMi backup to Server C.
  - d On Server C, run the `nnmrestore.ovpl -force -source temporary_location` command to complete a full NNMi restore.

For more information about which command options to use, see “NNMi Backup and Restore Tools” in the *NNMi Deployment Reference* and the `nnmrestore.ovpl` reference page, or the UNIX manpage.



Use the command options that match the backup you completed in [step 2](#)

- e On Server C, compare the `nms-local.properties.save` file from the temporary location to the `nms-local.properties` file located in the following directory: `%NNM_CONF%\nrm\props\`

Resolve any port conflicts, making changes to the `nms-local.properties` located in the above directory. Make sure to keep the `nmsas.server.port.web.http` (NNMi web server port) and `nmsas.server.port.web.https` (NNMi HTTPS web server port) values that were chosen during the NNMi installation on Server C.

- f Restart NNMi:

**ovstop**

**ovstart**

- 7 NNMi associates its license keys with a server's IP address. If the IP address for Server C is different from the IP address of Server A, obtain and install new NNMi license keys. See "Changing the IP Address of a Standalone NNMi Management Server" in the *NNMi Deployment Reference*.
- 8 On Server C, install NNMi 9.20.



# Moving NNMi from a RHEL Version below 5.4 to RHEL 5.4 or Greater

NNMi 9.20 does not support Red Hat Enterprise Linux (RHEL) versions below version 5.4. You must change the operating system to RHEL version 5.4 or greater before migrating to NNMi 9.20.

Use the information in this chapter if you have NNMi 9.0x or NNMi 9.1x (latest patch) running on a RHEL server (below version 5.4), and need to change the operating system to RHEL version 5.4 or greater.

This chapter contains the following topic:

[Changing NNMi from RHEL \(Versions below 5.4\) to RHEL Version 5.4 or Greater](#)

---

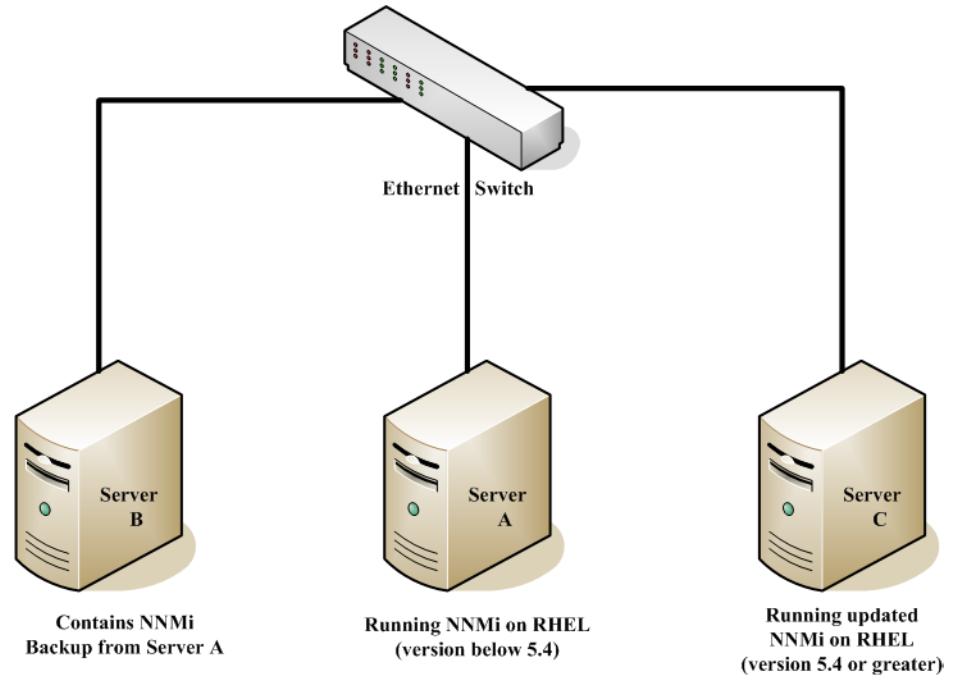
## Changing NNMi from RHEL (Versions below 5.4) to RHEL Version 5.4 or Greater

To complete the following steps, you must have NNMi 9.0x or NNMi 9.1x (latest patch) running on a RHEL server (version below 5.4). To check the NNMi version number, note the current patch level in the **Help->About HP Network Node Manager i Software** window. Verify that the version is 9.01.005 or later, or 9.11.003 or later. If the version is earlier than that, do not proceed. You need to install NNMi 9.0x or NNMi 9.1x (latest patch) before proceeding. See [Upgrading from NNMi 9.0x or 9.1x](#) on page 113 for the latest supported patch numbers.

To change an NNMi management server running NNMi 9.0x or NNMi 9.1x (latest patch) from RHEL (version below 5.4) to RHEL version 5.4 or greater, follow these steps:

- 1 Identify three servers that you will use during this procedure:
  - Server A is the current NNMi management server running RHEL (version below 5.4).
  - Server B will hold the NNMi backup files.
  - Server C will become the new NNMi management server running RHEL 5.4 or greater. This NNMi management server can be the same hardware as the current Server A.

Make sure the `hosts` file on the new NNMi management server contains the following entry: `127.0.0.1 localhost`



- 2 On Server A, run the `nnmbackup.ovpl -type online -scope all -target temporary_location` command to complete a full NNMi backup.

For more information about which command options to use, see “NNMi Backup and Restore Tools” in the *NNMi Deployment Reference* and the `nnmbackup.ovpl` reference page, or the UNIX manpage.

- 3 On Server A, copy the backup you completed in [step 2](#) to Server B.
- 4 On Server C, install RHEL 5.4 or greater.

▶ As an alternative to using Server C, reformat the disk on Server A and install RHEL 5.4 or greater. If you do that, substitute Server A for Server C for the remaining steps.

- 5 On Server C, install the same NNMi version as what was running on Server A. You must install the same patch level that NNMi Server A was at during the backup you completed in [step 2](#).

▶ You cannot back up with NNMi 9.0x and restore with NNMi 9.1x.

- 6 During the NNMi installation on Server C, the installation script might assign ports that differ from the Server B configuration. During the configuration restore on Server C, this might create port conflicts. To remedy this, do the following:
  - a On Server C, navigate to the following directory: `$NNM_CONF/nmm/props/`
  - b On Server C, copy the `nms-local.properties` file to `nms-local.properties.save` in a temporary location.
  - c On Server B, copy the NNMi backup to Server C.
  - d On Server C, run the `nnmrestore.ovpl -force -source temporary_location` command to complete a full NNMi restore.

For more information about which command options to use, see “NNMi Backup and Restore Tools” in the *NNMi Deployment Reference* and the *nmmrestore.ovpl* reference page, or the UNIX manpage.



Use the command options that match the backup you completed in [step 2](#).

- e On Server C, compare the `nms-local.properties.save` file from the temporary location to the `nms-local.properties` file located in the following directory: `$NNM_CONF/nmm/props/`

Resolve any port conflicts, making changes to the `nms-local.properties` located in the above directory. Make sure to keep the `jboss.http.port` (NNMi web server port) and `jboss.https.port` (NNMi HTTPS web server port) values that were chosen during the NNMi installation on Server C.

- f Restart NNMi:

**ovstop**

**ovstart**

- 7 NNMi associates its license keys with a server’s IP address. If the IP address for Server C is different from the IP address of Server A, obtain and install new NNMi license keys. See “Changing the IP Address of a Standalone NNMi Management Server” in the *NNMi Deployment Reference*.
- 8 On Server C, install NNMi 9.20.





# Migrating NNMi Oracle Data

If you plan to move the Oracle data in NNMi to Oracle 11G. The information in this chapter explains the steps to take to complete this work.

---

## Migrating NNMi Oracle Data

Suppose you have NNMi running in one of the following configurations:

- NNMi 9.0x with the latest patch connected to an Oracle 10G database and you need to upgrade to NNMi 9.20.
- NNMi 9.0x with the latest patch connected to an Oracle 11G database and you need to upgrade to NNMi 9.20.

The Oracle database instance migration you need to complete could include combinations of the following requirements:

- The existing Oracle instance running on NNMi 9.10 can be running Oracle 10G or 11G.
- The new Oracle instance running on NNMi 9.20 must be running Oracle 11G.
- The new Oracle instance can be located on the original server or on a different server and hostname.

To complete the migration of the NNMi Oracle data, complete the following steps:

- 1 As root or administrator, run the following command to stop NNMi: **ovstop -c**.
- 2 Use Oracle tools to move or copy the NNMi data from the existing Oracle server to the new server. Refer to Oracle documentation for additional information.



This Oracle data migration can be an in-place upgrade from Oracle 10 to Oracle 11 on the same server. Oracle provides database migration tools for converting Oracle 10 data into the Oracle 11 format.

- 3 *Only complete this step if the new Oracle server has a different hostname than the previous Oracle server.* On the NNMi management server, reconfigure NNMi to point to the new Oracle server by completing the following steps:

- a Edit the datasource configuration file shown below:



It is important that you complete the following steps accurately, or jboss will not correctly connect to the Oracle 11G database.

— *Windows*: %NNM\_JBOSS%\server\nms\deploy\nms-ds.xml

— *UNIX*: \$NNM\_JBOSS/server/nms/deploy/nms-ds.xml

- b Change the following attribute to reflect your new server

OLD:

```
<connection-url>jdbc:oracle:thin:@EXISTING_FQDN:EXISTING_ORACLE_PORT:EXISTING_SID </connection-url>
```

NEW:

```
<connection-url>jdbc:oracle:thin:@NEW_FQDN:NEW_PORT:NEW_SID</connection-url>
```

- 4 Complete one of the following actions:

If you are upgrading from NNMi 9.0x or NNMi 9.1x to NNMi 9.20, perform that migration now, following the installation instructions in the *HP Network Node Manager i Software Installation Guide*.

If you are already using NNMi 9.20, follow these steps to restart NNMi and complete the Oracle database move/migration:

- a Run the following command on the NNMi management server to restart NNMi: **ovstart -c**
- b Run the following command on the NNMi management server to check if all of the services are started and operating correctly: **ovstatus -v**

# Additional Upgrade Information

This chapter describes some changes between NNMi 9.20 and earlier NNMi versions. This chapter contains the following topics:

- [Configuration Differences](#)
- [MIBs](#)
- [Functionality Differences](#)

---

## Configuration Differences

- User groups replace NNMi roles for limiting user access within the NNMi console. User accounts can be mapped to multiple user groups.
  - For signing in to the NNMi console, each user account must be mapped to at least one of the NNMi-provided user groups. These groups are equivalent to the function of the NNMi role in previous releases.
  - In a multi-tenancy environment, each user account can be mapped to one or more custom user groups that provide access to a subset of the topology objects.

For more information, see “NNMi Security and Multi-Tenancy” in the *NNMi Deployment Reference*.

- The NNMi integration for retrieving user information from a directory service can now retrieve multiple group names per user.
  - For configuration option 2 (only user names and passwords in the directory service), existing integrations with a directory service continue to work without modification to the `ldap.properties` configuration file.
  - For configuration option 3 (all user information in the directory service), the following information applies:
    - In a single tenant environment (all NNMi console users can access all topology objects), existing integrations with a directory service continue to work without modification to the `ldap.properties` configuration file.

If you add any new NNMi user groups in the directory service, you must update the `ldap.properties` configuration file to the new model for retrieving user information from a directory service.

- In a multi-tenancy environment, update the `ldap.properties` configuration file to the new model for retrieving user information from a directory service.
- For information about updating the `ldap.properties` configuration file, see “Changing the Directory Service Access Configuration to Support the NNMi Security Model” in the *NNMi Deployment Reference*.
- NNMi 9.20 deprecates the following `ldap.properties` configuration file parameters. They will become unsupported in a future release:
  - `roleAttributeID`
  - `roleAttributeIsDN`
  - `roleNameAttributeID`
- After upgrading to NNMi 9.20, the following security and multi-tenancy configuration applies:
  - All nodes are assigned to the Default Tenant and the Default Security Group.
  - All users have access to all nodes in the NNMi topology and to all incidents.

This default configuration matches the object access available in NNMi 9.1x. For information on customizing object access, see “NNMi Security and Multi-Tenancy” in the *NNMi Deployment Reference*.
- If the HP NNMi—HP NA integration was configured on a NNMi 9.0x management server, the process of upgrading to NNMi 9.20 disables the configuration. For more information, see the *NNMi—Network Automation Integration Guide*.

## Application Failover

NNMi 9.0x supported either a UDP or a TCP solution for the application failover feature. NNMi 9.20 only supports the TCP solution. If you used the UDP application failover solution for NNMi 9.0x, and are upgrading to NNMi 9.20, the upgrade script converts your application failover configuration to the TCP solution. You must add the hostnames of all nodes in the cluster to the `com.hp.ov.nms.cluster.member.hostnames` parameter in the `nms-cluster.properties` file. For more information see “Configuring NNMi for Application Failover” in the *NNMi Deployment Reference*.

For the application failover feature to function correctly, the active and standby servers must have unrestricted network access to each other. NNMi 9.20 includes some port changes, so you might need to modify your firewall configuration. For more information see “NNMi 9.20 and Well-Known Ports” in the *NNMi Deployment Reference*.

## MIBs

If you loaded additional MIBs into earlier versions of NNMi that are not standards compliant or have dependencies on other MIB files, they might not migrate successfully. If a MIB does not migrate successfully, the trap configurations continue to work, however you might not be able to browse that MIB as you could prior to the migration.

If you suspect that some MIBs did not migrate, check the following directory for a failed subdirectory that contains the MIB file or files, failure details, and a log file with a name that associates it with the MIB file or files:

- *Windows*: %NNM\_DATA%\tmp\nnm9xMibMigrate
- *UNIX*: \$NNM\_DATA/tmp/nnm9xMibMigrate

Use the files contained in the above directories to determine why the MIBs did not migrate, then reload those MIBs.

---

## Functionality Differences

To review information about new features included in NNMi 9.20, see the “What’s New In This Version” section of the *NNMi Release Notes*.



# Upgrading Global and Regional Managers from NNMi 9.0x or 9.1x

---

## NNMi Versions Supported by Global Network Management

If a global manager is connected to a regional manager running NNMi 9.0x patch 2 or earlier, SNMP queries between the global and regional manager do not work. To remedy this, upgrade the regional manager to NNMi 9.0x patch 3 or later. To achieve the best results, the global manager must be the same version and NNMi patch level as the regional manager.



HP does not support a regional manager running NNMi 9.0x or 9.1x connected to a global manager running NNMi 9.20. The global manager and regional managers must be running the same NNMi version.

---

## Global Network Management Upgrade Steps

When upgrading NNMi management servers configured in a global network management environment to NNMi 9.20, the connections between the global manager and regional managers will drop until both the global and regional managers are upgraded to 9.20. For this reason, HP recommends you upgrade all of the servers at approximately the same time to minimize the total downtime.

For example, you might upgrade the NNMi management servers using the following steps:

- 1 Upgrade the regional managers to NNMi 9.20 and ensure proper operation. The global manager stays disconnected during the regional upgrades.

- 2 Upgrade the global manager to NNMi 9.20. The global manager performs a full resynchronization to obtain all events that occurred while the connection between the global manager and the regional managers was down. The effect is the same as if the administrator were to issue **`nnmnode rediscover.ovpl -all -fullsync`** from the global manager. See the *nnmnode rediscover.ovpl* reference page or the UNIX manpage for more information.



Note the following:

- Updates to Status and Incidents could be delayed as NNMi resynchronizes following an upgrade.
- If you see the following message during this resynchronization, it does not indicate a problem:

The Causal Engine's large queue size is causing delayed updates to Status and Incidents. This could be due to resynchronization following an upgrade, application failover, restore from backup, or a manual resynchronization.

- Do not stop NNMi during this resynchronization. To ensure resynchronization has completed, keep NNMi running for several hours following the upgrade.



# Application Failover and Upgrading to NNMi 9.20

---

## Application Failover and Upgrading from NNMi 9.0x or 9.1x

If you plan to upgrade an earlier version of NNMi that is running in an NNMi application failover configuration, follow the steps in the appropriate section below based on the database you are using.

### Embedded Database

To upgrade NNMi management servers configured for application failover and using the embedded database, follow these steps:

- 1 As a precaution, run the **nnmconfigexport.ovpl** script on both the active and standby NNMi management servers before proceeding. For information, see “Best Practice: Save the Existing Configuration” in the *NNMi Deployment Reference*.

As a precaution, back up your NNMi data on both the active and standby NNMi management servers before proceeding. For information, see “Backup Scope” in the *NNMi Deployment Reference*.

- 2 Complete the following steps on the active NNMi management server. Note that NNMi must be running for the following **nnmcluster** steps to work. Completing these steps will speed up the standby NNMi management server startup shown in [step 6](#) on page 138:

- a Run the **nnmcluster** command.

- b After NNMi prompts you, type **dbsync**, then press Enter. Review the displayed information to make sure it includes the following messages:

ACTIVE\_DB\_BACKUP: This means that the active NNMi management server is performing a new backup.

ACTIVE\_NNM\_RUNNING: This means that the active NNMi management server completed the backup referred to by the previous message.

STANDBY\_RECV\_DBZIP: This means that the standby NNMi management server is receiving a new backup from the active NNMi management server.

STANDBY\_READY: This means that the standby NNMi management server is ready to perform if the active NNMi management server fails.

- c Run **exit** or **quit** to stop the interactive `nmcluster` process you started in step a.
- 3 Run the **nmcluster -shutdown** command on the standby NNMi management server. This shuts down all `nmcluster` processes on the standby NNMi management server.
  - 4 To verify there are no `nmcluster` nodes running on the standby NNMi management server, *complete the following steps on the standby NNMi management server.*
    - a Run the **nmcluster** command.
    - b Verify that there are no (LOCAL) `nmcluster` nodes present except the one marked (SELF). There might be one or more (REMOTE) nodes present.
    - c Run **exit** or **quit** to stop the interactive `nmcluster` process you started in step a.
  - 5 *Complete the following steps on the standby NNMi management server to temporarily disable application failover:*
    - a Edit the following file:
      - Windows: `%NNM_SHARED_CONF%\props\nms-cluster.properties`
      - UNIX: `$NNM_SHARED_CONF/props/nms-cluster.properties`
    - b Comment out the `com.hp.ov.nms.cluster.name` parameter.
    - c Save your changes.
  - 6 Start, then stop processes on the standby NNMi management server.
    - a Run the **ovstart** command on the standby NNMi management server. Running the **ovstart** command causes the standby NNMi management server to import the transaction logs from the active NNMi management server.
    - b After the **ovstart** command completes, run the **ovstatus -v** command. All NNMi services should show the state `RUNNING`.
    - c Run the **ovstop** command on the standby NNMi management server.
  - 7 Upgrade the standby NNMi management server to NNMi 9.20 using the instructions located in the *NNMi Installation Guide*.



You must upgrade all of the iSPIs that you have installed on the standby NNMi management server to iSPI versions that support NNMi 9.20.

You now have the former active NNMi management server running NNMi 9.0x or 9.1x and the former standby NNMi management server running NNMi 9.20. You have both of these NNMi management servers running independently with no database synchronization. That means you have both NNMi management servers monitoring the network in parallel. Do not leave these NNMi management servers in this configuration for more than a few hours, as this configuration is a violation of the non-production license installed on the former standby node.

To complete the upgrade, and remedy this situation, select a time to upgrade the former active node to NNMi 9.20. Have the operators temporarily use the former standby node to monitor the network while you complete the upgrade.

The remainder of this procedure assumes you plan to retain the database information from the former active node and discard the database information from the former standby node.

- 8 Run the **nnmcluster -halt** command on the former active NNMi management server.
- 9 To verify there are no **nnmcluster** nodes running on the former active NNMi management server, *complete the following steps on the former active NNMi management server.*
  - a Run the **nnmcluster** command.
  - b Verify that there are no (LOCAL) **nnmcluster** nodes present except the one marked (SELF). There might be one or more (REMOTE) nodes present.
  - c Run **exit** or **quit** to stop the interactive **nnmcluster** process you started in step a.
- 10 *Complete the following steps on the former active NNMi management server to temporarily disable application failover:*
  - a Edit the following file:
    - Windows: %NNM\_SHARED\_CONF%\props\nms-cluster.properties
    - UNIX: \$NNM\_SHARED\_CONF/props/nms-cluster.properties
  - b Comment out the `com.hp.ov.nms.cluster.name` parameter.

Upgrade the former active NNMi management server to NNMi 9.20 using the instructions located in the *NNMi Installation Guide*.



You must upgrade all of the iSPIs that you have installed on the former active NNMi management server to iSPI versions that support NNMi 9.20.

Now you have two servers running NNMi 9.20, but they are still independent since the databases are not synchronized.

- 11 Complete the following steps on the former active NNMi management server:
  - a Run the **ovstop** command.
  - b Edit the following file:
    - Windows: %NNM\_SHARED\_CONF%\props\nms-cluster.properties
    - UNIX: \$NNM\_SHARED\_CONF/props/nms-cluster.properties
  - c Type in the value of the `com.hp.ov.nms.cluster.name` parameter.



The NNMi upgrade procedure does not preserve commented-out properties. Therefore, you must retype the cluster name.

- d Uncomment the `com.hp.ov.nms.cluster.name` parameter.
  - e Save your changes.
- 12 Run either the **ovstart** or **nnmcluster -daemon** command on the former active NNMi management server. It is now the active node.
- 13 Instruct the operators to begin using the active node to monitor the network.



The former standby NNMi management server discards all of the database activity occurring during the maintenance window, from [step 8](#) on page 139 through [step 12](#) on page 139.

- 14 Complete the following steps on the former standby NNMi management server:
  - a Run the **ovstop** command.
  - b Edit the following file:
    - *Windows*: %NNM\_SHARED\_CONF%\props\nms-cluster.properties
    - *UNIX*: \$NNM\_SHARED\_CONF/props/nms-cluster.properties
  - c Type in the value of the `com.hp.ov.nms.cluster.name` parameter.
  - d Uncomment the `com.hp.ov.nms.cluster.name` parameter.
  - e Save your changes.
- 15 Run either the **ovstart** or **nnmcluster -daemon** command on the former standby NNMi management server.

This NNMi management server becomes the standby node, and receives a copy of the database from the active node.

- 16 If you installed the NNM iSPI Performance for QA, the NNM iSPI Performance for Metrics, or the NNM iSPI Performance for Traffic; are using the application failover feature; and completed the upgrade process shown above, run the NNM iSPI enablement script for each NNM iSPI on both the active and standby NNMi management servers. The path to the NNM iSPI enablement script is as follows:
  - *Windows*: %NNMInstallDir%\bin\nmenableperfspi.ovpl
  - *UNIX*: /opt/OV/bin/nmenableperfspi.ovpl

## Oracle Database



You must upgrade NNMi management servers separately because two NNMi management servers cannot be simultaneously connected to the same Oracle database.

To upgrade NNMi management servers configured for application failover and using an Oracle database, follow these steps:

- 1 As a precaution, run the **nnmconfigexport.ovpl** script on both the active and standby NNMi management servers before proceeding. For information, see "Best Practice: Save the Existing Configuration" in the *NNMi Deployment Reference*.
- 2 As a precaution, back up your NNMi data on both the active and standby NNMi management servers before proceeding. For information, see "Backup Scope" in the *NNMi Deployment Reference*.
- 3 Run the **nnmcluster -halt** command on the standby NNMi management server. This shuts down all **nnmcluster** processes on both the active and standby NNMi management server.
- 4 To verify there are no **nnmcluster** nodes running on either the active or standby NNMi management server, *complete the following steps on the standby NNMi management server*.
  - a Run the **nnmcluster** command.
  - b Verify that the only **nnmcluster** node present is one marked (SELF).
  - c Run **exit** or **quit** to stop the interactive **nnmcluster** process you started in step a.

- 5 Complete the following steps on the standby NNMi management server to temporarily disable application failover:

- a Edit the following file:
  - *Windows:* %NNM\_SHARED\_CONF%\props\nms-cluster.properties
  - *UNIX:* \$NNM\_SHARED\_CONF/props/nms-cluster.properties
- b Comment out the `com.hp.ov.nms.cluster.name` parameter.
- c Save your changes.

- 6 Upgrade the standby NNMi management server to NNMi 9.20 using the instructions located in the *NNMi Installation Guide*.



You must upgrade all of the iSPIs that you have installed on the standby NNMi management server to iSPI versions that support NNMi 9.20.

You now have the former standby NNMi management server with NNMi 9.20 installed, and the former active NNMi management server with NNMi 9.0x or 9.1x installed.

- 7 Run the **ovstop** command on the former standby NNMi management server to disconnect the NNMi management server from the Oracle database.
- 8 Complete the following steps on the former active NNMi management server to temporarily disable application failover:

- a Edit the following file:
  - *Windows:* %NNM\_SHARED\_CONF%\props\nms-cluster.properties
  - *UNIX:* \$NNM\_SHARED\_CONF/props/nms-cluster.properties
- b Comment out the `com.hp.ov.nms.cluster.name` parameter.

- 9 Upgrade the former active NNMi management server to NNMi 9.20 using the instructions located in the *NNMi Installation Guide*.



You must upgrade all of the iSPIs that you have installed on the former active NNMi management server to iSPI versions that support NNMi 9.20.

Now you have two servers with NNMi 9.20 installed.

- 10 Complete the following steps on the former active NNMi management server:

- a Run the **ovstop** command.
- b Edit the following file:
  - *Windows:* %NNM\_SHARED\_CONF%\props\nms-cluster.properties
  - *UNIX:* \$NNM\_SHARED\_CONF/props/nms-cluster.properties
- c Type in the value of the `com.hp.ov.nms.cluster.name` parameter.



The NNMi upgrade procedure does not preserve commented-out properties. Therefore, you must retype the cluster name.

- d Uncomment the `com.hp.ov.nms.cluster.name` parameter.
- e Save your changes.

- 11 Run the **ovstart** or **nnmcluster -daemon** command on the former active NNMi management server. It is now the active node.

- 12 Complete the following steps on the former standby NNMi management server:

- f Edit the following file:
    - *Windows*: %NNM\_SHARED\_CONF%\props\nms-cluster.properties
    - *UNIX*: \$NNM\_SHARED\_CONF/props/nms-cluster.properties
  - g Type in the value of the `com.hp.ov.nms.cluster.name` parameter.
  - h Uncomment the `com.hp.ov.nms.cluster.name` parameter.
  - i Save your changes.
- 13 Run either the **ovstart** or **nnmcluster -daemon** command on the former standby NNMi management server.

This NNMi management server becomes the standby node.

- 14 If you installed the NNM iSPI Performance for QA, the NNM iSPI Performance for Metrics, or the NNM iSPI Performance for Traffic; are using the application failover feature; and completed the upgrade process shown above, run the NNM iSPI enablement script for each NNM iSPI on both the active and standby NNMi management servers. The path to the NNM iSPI enablement script is as follows:
- *Windows*: %NNMInstallDir%\bin\nmenableperfspi.ovpl
  - *UNIX*: /opt/OV/bin/nmenableperfspi.ovpl

## Application Failover and NNMi Patches

Both NNMi management servers must be running the same NNMi version and patch level. To add patches to the active and standby NNMi management servers, use one of the following procedures:

- [Applying Patches for Application Failover \(Shut Down Both Active and Standby\)](#)  
Use this procedure when you are not concerned with an interruption in network monitoring.
- [Applying Patches for Application Failover \(Keep One Active NNMi Management Server\)](#)  
Use this procedure when must avoid any interruptions in network monitoring.

### Applying Patches for Application Failover (Shut Down Both Active and Standby)

This procedure results in both NNMi management servers being non-active for some period of time during the patch process. To apply patches to the NNMi management servers configured for application failover, follow these steps:

- 1 As a precaution, run the **nnmconfigexport.ovpl** script on both the active and standby NNMi management servers before proceeding. For information, see "Best Practice: Save the Existing Configuration" in the *NNMi Deployment Reference*.
- 2 As a precaution, back up your NNMi data on both the active and standby NNMi management servers before proceeding. For information, see "Backup Scope" in the *NNMi Deployment Reference*.
- 3 As a precaution, on the active NNMi management server, do the following steps:
  - a Run the **nnmcluster** command.

- b Embedded database only: After NNMi prompts you, type **dbsync**, then press Enter. Review the displayed information to make sure it includes the following messages:

ACTIVE\_DB\_BACKUP: This means that the active NNMi management server is performing a new backup.

ACTIVE\_NNM\_RUNNING: This means that the active NNMi management server completed the backup referred to by the previous message.

STANDBY\_READY: This shows the previous status of the standby NNMi management server.

STANDBY\_RECV\_DBZIP: This means that the standby NNMi management server is receiving a new backup from the active NNMi management server.

STANDBY\_READY: This means that the standby NNMi management server is ready to perform if the active NNMi management server fails.

- 4 Run the **nnmcluster -halt** command on the active NNMi management server. This shuts down all **nnmcluster** processes on both the active and standby NNMi management servers.
- 5 To verify there are no **nnmcluster** nodes running on either server, *complete the following steps on both the active and standby NNMi management servers.*
  - a Run the **nnmcluster** command.
  - b Verify that there are no **nnmcluster** nodes present except the one marked (SELF).
  - c Run **exit** or **quit** to stop the interactive **nnmcluster** process you started in step a.
- 6 On the active NNMi management server, comment out the `com.hp.ov.nms.cluster.name` parameter in the `nms-cluster.properties` file.
  - a Edit the following file:
    - *Windows:* `%NNM_SHARED_CONF%\props\nms-cluster.properties`
    - *UNIX:* `$NNM_SHARED_CONF/props/nms-cluster.properties`
  - b Comment out the `com.hp.ov.nms.cluster.name` parameter.
  - c Save your changes.
- 7 Apply the NNMi patch to the active NNMi management server using the instructions provided with the patch.
- 8 On the active NNMi management server, uncomment the `com.hp.ov.nms.cluster.name` parameter in the `nms-cluster.properties` file.
  - a Edit the following file:
    - *Windows:* `%NNM_SHARED_CONF%\props\nms-cluster.properties`
    - *UNIX:* `$NNM_SHARED_CONF/props/nms-cluster.properties`
  - b Uncomment the `com.hp.ov.nms.cluster.name` parameter.
  - c Save your changes.
- 9 Run the **ovstart** command on the active NNMi management server.
- 10 Verify that the patch installed correctly on the active NNMi management server by viewing information on the **Product** tab of the **Help > System Information** window in the NNMi console.
- 11 Run the **nnmcluster -dbsync** command to create a new backup.



- 12 On the standby NNMi management server, comment out the `com.hp.ov.nms.cluster.name` parameter in the `nms-cluster.properties` file as shown in [step a](#) on page 143 through [step c](#) on page 143.
- 13 Apply the NNMi patch to the standby NNMi management server.
- 14 On the standby NNMi management server, uncomment the `com.hp.ov.nms.cluster.name` parameter in the `nms-cluster.properties` file as shown in [step a](#) on page 143 through [step c](#) on page 143.
- 15 Run the **ovstart** command on the standby NNMi management server.
- 16 If you installed the NNM iSPI Performance for QA, the NNM iSPI Performance for Metrics, or the NNM iSPI Performance for Traffic; are using the application failover feature; and completed the patch process shown above, run the NNM iSPI enablement script for each NNM iSPI on both the active and standby NNMi management servers.

### Applying Patches for Application Failover (Keep One Active NNMi Management Server)

This procedure results in one NNMi management server always being active during the patch process.



This process results in continuous monitoring of the network, however NNMi loses the transaction logs occurring during this patch process.

To apply NNMi patches to the NNMi management servers configured for application failover, follow these steps:

- 1 As a precaution, run the **nnmconfigexport.ovpl** script on both the active and standby NNMi management servers before proceeding. For information, see "Best Practice: Save the Existing Configuration" in the *NNMi Deployment Reference*.
- 2 As a precaution, back up your NNMi data on both the active and standby NNMi management servers before proceeding. For information, see "Backup Scope" in the *NNMi Deployment Reference*.
- 3 Run **nnmcluster** on one of the nodes.
- 4 Enter **dbsync** on the NNMi management server used in the previous step to synchronize the two databases.



The **dbsync** option works on an NNMi management server using the embedded database. Do not use the **dbsync** option on an NNMi management server configured to use an Oracle database.

- 5 Wait until the active NNMi management server reverts to `ACTIVE_NNM_RUNNING` and the standby NNMi management server reverts to `STANDBY_READY`, before continuing.
- 6 Exit or quit from the **nnmcluster** command.
- 7 Stop the cluster on the standby NNMi management server by running the following command on the standby NNMi management server:  
**nnmcluster -shutdown**
- 8 Make sure the following processes and services terminate before continuing:
  - postgres
  - ovjboss



- 9 Make sure the `nnmcluster` process terminates before continuing. If the `nnmcluster` process will not terminate, manually kill the `nnmcluster` process only as a last resort.
- 10 Edit the following file on the standby NNMi management server:
  - Windows:* %nnmDataDir%\shared\nnm\conf\props\nms-cluster.properties
  - UNIX:* \$nnmDataDir/shared/nnm/conf/props/nms-cluster.properties
- 11 Comment out the cluster name by placing a `#` at the front of the line, then save your changes:
  - #com.hp.ov.nms.cluster.name = NNMiCluster**
- 12 Install the NNMi patch on the standby NNMi management server.
- 13 At this point, the standby NNMi management server is patched but stopped, and the active NNMi management server is unpatched but running. Stop the active NNMi management server and immediately bring the standby NNMi management server online to monitor your network.
- 14 Shut down the cluster on the active NNMi management server by running the following command on the active NNMi management server:
  - nnmcluster -halt**
- 15 Make sure the `nnmcluster` process terminates. If it does not terminate within a few minutes, manually kill the `nnmcluster` process.
- 16 On the standby NNMi management server, uncomment the cluster name from the `nms-cluster.properties` file.
- 17 Start the cluster on the standby NNMi management server by running the following command on the standby NNMi management server:
  - nnmcluster -daemon**
- 18 Install the NNMi patch on the active NNMi management server.
- 19 At this point, the previous active NNMi management server is patched but offline. Bring it back into the cluster (as the standby NNMi management server) by performing the following:
  - a Uncomment the entry in the `nms-cluster.properties` file on the active NNMi management server.
  - b Start the active NNMi management server using the following command:
    - nnmcluster -daemon**
- 20 To monitor the progress, run the following command on both the active and standby NNMi management servers:
  - nnmcluster**

Wait until the previous active NNMi management server finishes retrieving the database from the previous standby NNMi management server.
- 21 After the previous active NNMi management server displays `STANDBY_READY`, run the following command on the previous active NNMi management server:
  - nnmcluster -acquire**
- 22 If you installed the NNM iSPI Performance for QA, the NNM iSPI Performance for Metrics, or the NNM iSPI Performance for Traffic; are using the application failover feature; and completed the patch process shown above, run the NNM iSPI enablement script for each NNM iSPI on both the active and standby NNMi management servers.



# High Availability and Upgrading from NNMi 9.0x or 9.1x to NNMi 9.20

Follow the appropriate procedure for your environment:

- [Upgrade NNMi with the Embedded Database on all Supported Operating Systems](#) on page 147
- [Upgrade NNMi with Oracle on all Supported Operating Systems](#) on page 151

## Upgrade NNMi with the Embedded Database on all Supported Operating Systems



As of NNMi 9.10, Serviceguard is no longer supported on the Linux operating system. If NNMi is currently running under Serviceguard HA, you cannot follow the procedure in this section. Instead, unconfigure NNMi from HA as described in [Unconfiguring NNMi from an HA Cluster](#) on page 151, upgrade NNMi on all nodes, and then configure NNMi to run under a supported HA product as described in "Configure NNMi for HA" in the *NNMi Deployment Reference*. Alternatively, you can configure NNMi for NNMi application failover as described in "Configuring NNMi for Application Failover" in the *NNMi Deployment Reference*.

Upgrading NNMi includes upgrading the Postgres database software to a newer version. For this reason, NNMi must be taken out of operation for the duration of the upgrade process.



NNMi will be unavailable for approximately 30 to 60 minutes during this upgrade procedure.

To upgrade from NNMi 9.0x or 9.1x under HA to NNMi 9.20 under HA, upgrade the active node to update the embedded database, and then upgrade the passive node while NNMi is still in maintenance mode. Follow these steps:

- 1 Ensure that the NNMi 9.0x or 9.1x configuration is consistent across all HA nodes by forcing a failover, in turn, to each of the passive nodes.
- 2 For NNMi 9.0x, ensure that all nodes are running NNMi 9.0x Patch 5 or a higher version. For NNMi 9.1x, use patch 3 or higher.

If necessary, upgrade each system to the appropriate consolidated patch.

- 3 Check the `ov.conf` files on both systems to ensure that they have the correct values. The `ov.conf` file is available in the following location:
  - *Windows*: `%NnmDataDir%\shared\nnm\conf`
  - *UNIX*: `$NnmDataDir/shared/nnm/conf`
- 4 Determine which node in the NNMi 9.0x or 9.1x HA cluster is active:
  - *Windows*:
 

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \
-group <resource_group> -activeNode
```
  - *UNIX*:
 

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \
-group <resource_group> -activeNode
```

The remainder of this procedure refers to the currently active node as server X and the currently passive node as server Y.
- 5 For HP-UX systems, on server Y, edit the `/etc/cmcluster/<resource_group>/<resource_group>.mon` file as follows:
  - a Locate the following line:
 

```
if [ ! -f /var/opt/OV/hacluster/$HA_RESOURCE_GROUP/maint_NNM -a
! -f /var/opt/OV/hacluster/$HA_RESOURCE_GROUP/maint_NNM ]
```
  - b Change the second "maint\_NNM" to "maintenance".
  - c Failover the application and repeat [step a](#) and [step b](#) on the node that is no longer running the resource group.
- 6 For Windows systems, perform the following:
  - a On server X, stop the `<resource_group>-app` resource.
  - b Check the Access Control Lists (ACLs) on the file `%NnmDataDir%\hacluster\<resource_group>\hamscs.vbs` (be sure to remember these).
  - c Save the `hamscs.vbs` file.
  - d Copy the `%NnmInstallDir%\misc\nnm\ha\nnmhamscs.vbs` script to a temporary directory where you can edit the file.
  - e Open the copy of the `nnmhamscs.vbs` file and change all references for `product_name` to be **NNM**. You can reference the original script for the value. Save the `nnmhamscs.vbs` file.
  - f As Administrator, copy the updated `nnmhamscs.vbs` script to `%NnmDataDir%\hacluster\<resource_group>\hamscs.vbs`.
  - g Check the ACLs again to ensure that they are the same as before.
  - h Start the `<resource_group>-app` resource.
  - i Verify that the resource comes online. If not, check the cluster logs to see if there are any syntax errors. (You can use the following command to generate a cluster log: `cluster log /gen`. If you must specify a folder, you can do so using the following syntax: `cluster log /gen /copy:<my folder>`.)
  - j Run `ovstop`.
- 7 On server X, disable HA resource group monitoring by creating the following maintenance file:

- *Windows:*

**%NnmDataDir%\hacluster\*<resource\_group>*\maintenance**



Ensure that the maintenance file does not have a .txt extension, which can occur if the file has been edited with a text editor, such as Notepad.

- *UNIX:*

**\$NnmDataDir/hacluster/*<resource\_group>*/maintenance**

The file can be empty.

- 8 On server X, upgrade NNMi:

- a Upgrade NNMi to the current version as described in this manual.

The database upgrade occurs during this step.

- b To verify that the upgrade completed correctly, enter the following command:

**ovstart**

All NNMi services should show the state RUNNING.

- c Upgrade all add-on NNM iSPIs to version 9.20.

For information, see the documentation for each NNM iSPI.



If your environment includes standalone NNM iSPIs, you must also upgrade those products to version 9.20 for correct functionality. You can complete those upgrades after completing this procedure.

- 9 For Windows systems, do the following:

- a Copy the updated `nmhamscs.vbs` script (see [step f](#) within [step 6](#)) from Server X to `%NnmDataDir%\hacluster\<resource_group>\hamscs.vbs` on Server Y.

- b Check the ACLs to ensure that they are the same as before.

- 10 On server X, run the following command: **nmhadisk.ovpl NNM -replicate**.

- 11 On server Y, disable HA resource group monitoring by creating the following maintenance file:

- *Windows:*

**%NnmDataDir%\hacluster\*<resource\_group>*\maintenance**



Ensure that the maintenance file does not have a .txt extension, which can occur if the file has been edited with a text editor, such as Notepad.

- *UNIX:*

**\$NnmDataDir/hacluster/*<resource\_group>*/maintenance**

The file can be empty.

- 12 On server Y, upgrade NNMi:

- a Upgrade NNMi to the current version as described in this manual.

- b Verify that the upgrade completed without error.

- c Upgrade all add-on NNM iSPIs to version 9.20.

For information, see the documentation for each NNM iSPI.

- 13 If the HA cluster includes multiple passive nodes, repeat [step 12](#) for each passive node.

- 14 For HP-UX systems, on the node not running the resource group, run the following commands:

```
cd /etc/cmcluster/<resource_group>
cp <resource_group>.mon <resource_group>.mon.save
cp /opt/OV/misc/nnm/ha/mcsg/NNM/rg.mon <resource_group>.mon
```

- 15 On server X, delete the maintenance file:

- *Windows:*  
%NnmDataDir%\hacluster\*<resource\_group>*\maintenance
- *UNIX:*  
\$NnmDataDir/hacluster/*<resource\_group>*/maintenance

- 16 Perform the following post-installation steps:

- a Verify that the following variables are set:

```
NNM_INTERFACE
HA_MOUNT_POINT
NNM_ADD_ON_PRODUCTS
HA_LOCALE (not required if running in C)
```

These variables are defined in the following locations:

*HP-UX Serviceguard:*

```
/etc/cmcluster/<resource_group>/<resource_group>.public.env
```

*Veritas:*

```
/opt/VRTSvcs/bin/hagrp -display | grep UserStrGlobal
```

*Windows:* Using regedit, the values are in the following location:

```
HKEY_LOCAL_MACHINE\Cluster\Groups\<group>\Parameters
```

- b If the variables are not set, you can run the following commands for each missing value:

```
/opt/OV/misc/nnm/ha/nnmhaclusterinfo.ovpl -config NNM -set
NNM_INTERFACE <value for NNM_INTERFACE>
```

```
/opt/OV/misc/nnm/ha/nnmhaclusterinfo.ovpl -config NNM -set
HA_MOUNT_POINT <value for HA_MOUNT_POINT>
```

```
/opt/OV/misc/nnm/ha/nnmhaclusterinfo.ovpl -config NNM -set
NNM_ADD_ON_PRODUCTS <value for NNM_ADD_ON_PRODUCTS>
```

```
/opt/OV/misc/nnm/ha/nnmhaclusterinfo.ovpl -config NNM -set
HA_LOCALE <value for HA_LOCALE>
```



HA\_LOCALE is only needed if you are attempting to use a localized language.

- 17 For all Linux HA upgrades, run the following sets of commands, as applicable for your system:

— *RHEL:*

```
rm /etc/rc.d/rc*.d/S98netmgt
```

```
rm /etc/rc.d/rc*.d/K01netmgt
```

— *SuSE:*

```
rm /etc/init.d/rc*.d/S98netmgt
```

```
rm /etc/init.d/rc*.d/K01netmgt
```



When using Windows Server 2008 R2, the Network Name resource may have the name "Network Name". This name should be the short name for the virtual IP address. If applicable, change the name as follows:

- 1 Using Failover Cluster Management, select the Network Name resource.
- 2 Right-click and select **Properties**.
- 3 Change the name.

## Upgrade NNMi with Oracle on all Supported Operating Systems

To upgrade NNMi for HA in an Oracle environment, follow the procedure described in [Upgrade NNMi with the Embedded Database on all Supported Operating Systems](#) on page 147.

---

## Unconfiguring NNMi from an HA Cluster

The process of removing an NNMi node from an HA cluster involves undoing the HA configuration for that instance of NNMi. You can then run that instance of NNMi as a standalone management server, or you can uninstall NNMi from that node.

If you want to keep NNMi configured for high availability, the HA cluster must contain one node that is actively running NNMi and at least one passive NNMi node. If you want to completely remove NNMi from the HA cluster, unconfigure the HA functionality on all nodes in the cluster.

To completely unconfigure NNMi from an HA cluster, follow these steps:

- 1 Determine which node in the HA cluster is active. On any node, run the following command:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \  
-group <resource_group> -activeNode
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \  
-group <resource_group> -activeNode
```

- 2 On each passive node, unconfigure any add-on NNM iSPIs from the HA cluster.

For information, see the documentation for each NNM iSPI.

- 3 On any node in the HA cluster, verify that the add-on NNM iSPIs on all passive nodes have been unconfigured from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \  
-config NNM -get NNM_ADD_ON_PRODUCTS
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \
-config NNM -get NNM_ADD_ON_PRODUCTS
```

The command output lists the add-on iSPI configurations in the format `<iSPI_PM_Name>[hostname_list]`. For example:

```
PerfSPIHA[hostname1, hostname2]
```

At this time, only the active node hostname should appear in the output. If a passive node hostname appears in the output, repeat [step 2](#) until this command output includes only the active node hostname.

- 4 On each passive node, unconfigure NNMi from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM \
<resource_group>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM \
<resource_group>
```

This command removes access to the shared disk but does not unconfigure the disk group or the volume group.

- 5 On each passive node, move the NNMi HA resource group-specific files to a separate location for safe-keeping:



If you do not plan to reconfigure the NNMi HA resource group, you do not need to save a copy of these files, and you can delete them at this time.

- *MSFC:* In Windows Explorer, delete the `%NnmDataDir%\hacluster\<resource_group>\` folder.

- *Serviceguard:*

```
rm -rf /var/opt/OV/hacluster/ <resource_group>
rm -rf /etc/cmcluster/ <resource_group>
```

- *VCS:*

```
rm -rf /var/opt/OV/hacluster/ <resource_group>
```

- *RHCS:*

```
rm -rf /var/opt/OV/hacluster/ <resource_group>
```

- 6 On the active node, unconfigure any add-on NNM iSPIs from the HA cluster.

For information, see the documentation for each NNM iSPI. On any node in the HA cluster, verify that the add-on NNM iSPIs on all nodes have been unconfigured from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \
-config NNM -get NNM_ADD_ON_PRODUCTS
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \
-config NNM -get NNM_ADD_ON_PRODUCTS
```



If any hostname appears in the output, repeat [step 6](#) until this command output indicates that no iSPIs are configured.

- 7 On the active node, stop the NNMi HA resource group:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhastoprg.ovpl NNM \  
<resource_group>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhastoprg.ovpl NNM \  
<resource_group>
```

This command does not remove access to the shared disk. Nor does it unconfigure the disk group or the volume group.

- 8 On the active node, unconfigure NNMi from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM \  
<resource_group>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM \  
<resource_group>
```

This command removes access to the shared disk but does not unconfigure the disk group or the volume group.

- 9 On the active node, move the NNMi HA resource group-specific files to a separate location for safe-keeping:



If you do not plan to reconfigure the NNMi HA resource group, you do not need to save a copy of these files, and you can delete them at this time.

- *MSFC:* In Windows Explorer, delete the  
%NnmDataDir%\hacluster\*<resource\_group>*\ folder.

- *Serviceguard:*

```
rm -rf /var/opt/OV/hacluster/ <resource_group>  
rm -rf /etc/cmcluster/ <resource_group>
```

- *VCS:*

```
rm -rf /var/opt/OV/hacluster/ <resource_group>
```

- *RHCS:*

```
rm -rf /var/opt/OV/hacluster/ <resource_group>
```

- 10 Unmount the shared disk.

- If you want to reconfigure the NNMi HA cluster at some point, you can keep the disk in its current state.
- If you want to use the shared disk for another purpose, copy all data that you want to keep (as described in [Running NNMi Outside HA with the Existing Database](#) on page 154), and then use the HA product commands to unconfigure the disk group and volume group.

## Running NNMi Outside HA with the Existing Database

If you want to run NNMi outside HA on any node with the existing database, follow these steps:

- 1 On the active node (if one still exists), ensure that NNMi is not running:

```
ovstop
```

Alternatively, check the status of the `ovspmd` process by using Task Manager (Windows) or the `ps` command (UNIX).

- 2 On the current node (where you want to run NNMi outside HA), verify that NNMi is not running:

```
ovstop
```



To prevent data corruption, make sure that no instance of NNMi is running and accessing the shared disk.

- 3 (UNIX only) Activate the disk group, for example, on HP-UX Serviceguard:

```
vgchange -a e <disk_group>
```

- 4 Use the appropriate operating system commands to mount the shared disk. For example:

- *Windows:* Use Server Manager—>Disk Management.
- *UNIX:* `mount /dev/vgnrm/lvnm /nnmmount`

- 5 Copy the NNMi files from the shared disk to the local disk:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM \  
-from <HA_mount_point>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM \  
-from <HA_mount_point>
```

- 6 Use the appropriate operating system commands to unmount the shared disk. For example:

- *Windows:* Use Windows Explorer.
- *UNIX:* `umount /nnmmount`

- 7 (UNIX only) Deactivate the disk group, for example:

```
vgchange -a n <disk_group>
```

- 8 Obtain and install the permanent production license keys for the physical IP address of this NNMi management server as described in the "Licensing NNMi" chapter in the *NNMi Deployment Reference*.

- 9 Start NNMi:

```
ovstart -c
```

NNMi is now running with a copy of the database that was formerly used by the NNMi HA resource group. Manually remove from the NNMi configuration any nodes that you do not want to manage from this NNMi management server.

## Patching NNMi under HA

To apply a patch for NNMi, work in HA maintenance mode. Follow these steps:

- 1 Determine which node in the HA cluster is active:
  - *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl \
-group <resource_group> -activeNode
```
  - *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl \
-group <resource_group> -activeNode
```
- 2 On the active node, put the NNMi HA resource group into maintenance mode as described in “Putting an HA Resource Group into Maintenance Mode” in the *NNMi Deployment Reference*.
 

Include the `NORESTART` keyword.
- 3 On all passive nodes, put the NNMi HA resource group into maintenance mode as described in “Putting an HA Resource Group into Maintenance Mode” in the *NNMi Deployment Reference*.
 

Include the `NORESTART` keyword.
- 4 On the active node, follow these steps:
  - a Stop NNMi:
 

```
ovstop -c
```
  - b Back up the shared disk by performing a disk copy.
  - c *Optional.* Use the `nnmbackup.ovpl` command, or another database command, to back up all NNMi data. For example:
 


```
nnmbackup.ovpl -type offline -scope all -target nnmi_backups
```

For more information about this command, see “NNMi Backup and Restore Tools” in the *NNMi Deployment Reference*.
  - d Apply the appropriate NNMi and NNM iSPI patches to the system.
  - e Start NNMi:
 

```
ovstart -c
```
  - f Verify that NNMi started correctly:
 

```
ovstatus -c
```

All NNMi services should show the state `RUNNING`.
- 5 On each passive node, apply the appropriate patches to the system.
 

 Never run the `ovstart` or `ovstop` commands on a secondary (backup) cluster node.
- 6 On all passive nodes, take the NNMi HA resource group out of maintenance mode as described in “Removing an HA Resource Group from Maintenance Mode” in the *NNMi Deployment Reference*.

- 7 On the active node, take the NNMi HA resource group out of maintenance mode as described in “Removing an HA Resource Group from Maintenance Mode” in the *NNMi Deployment Reference*.

# We appreciate your feedback!

If an email client is configured on this system, by default an email window opens when you click *here*.

If no email client is available, copy the information below to a new message in a web mail client, and then send this message to **ovdoc-nsm@hp.com**.

**Product name and version:** NNMi 9.20

**Document title:** *NNMi Upgrade Reference*

**Feedback:**

