

HP Network Node Manager i Software

For the Windows[®], HP-UX, Linux, and Solaris operating systems

Software Version: 9.0x Patch 1

Deployment Reference

Document Release Date: June 2010
Software Release Date: June 2010



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 Notices

© Copyright 2008–2010 Hewlett-Packard Development Company, L.P.

Trademark Notices

Acrobat® 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.

Java™ is a US trademark of Sun Microsystems, Inc.

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

Oracle is a registered trademark of Oracle Corporation 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 USA, 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>)

This product includes software developed by The Legion Of The Bouncy Castle.
(<http://www.bouncycastle.org>)

This product contains software developed by Trantor Standard Systems Inc..
(<http://www.trantor.ca>)

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 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

You can visit the HP software support web site at:

www.hp.com/go/hpsoftwaresupport

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

HP Software Support Online 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 HP Software Support web site to:

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

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

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

To find more information about access levels, go to:

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

Contents

About This Guide	23
What Is in This Guide?	23
Environment Variables Used in This Document	24
Revision History	25
For More Information about NNMi	26
Preparation	29
Hardware and Software Requirements	31
Supported Hardware and Software	31
System Configuration (UNIX)	32
NNMi Coexistence with HP Performance Insight	33
NNMi Coexistence with HP Operations Agent	33
Configuration	35
General Concepts for Configuration	37
Task Flow Model	37
Best Practice: Save the Existing Configuration	38
Best Practice: Use the Author Attribute	38
User Interface Model	38
Ordering	38
Node Groups	39
Group Overlap	40
Node Group Membership	41
Hierarchies/Containment	41
Device Filters	41
Additional Filters	42
Additional Nodes	42
Node Group Status	42
Interface Groups	43
Node/Interface/Address Hierarchy	43
Stop Everything and Start Over Again	44
NNMi Communications	45
Concepts for Communications	46
Levels of Communication Configuration	46
Network Latency and Timeouts	46
SNMP Access Control	47
SNMP Version Preferences	48

Management Address Preferences	49
Polling Protocols	49
Plan Communications	50
Default Communication Settings	50
Communication Configuration Regions	50
Specific Node Configurations	51
Retry and Timeout Values	52
Active Protocols	52
Multiple Community Strings or Authentication Profiles	52
Configure Communications	53
Evaluate Communications	53
Are All Nodes Configured for SNMP?	53
Is SNMP Access Currently Available for a Device?	54
Is the Management IP Address Correct?	54
Is NNMi Using the Correct Communications Settings?	54
Do the State Poller Settings Agree with the Communication Settings?	54
Tune Communications	55
NNMi Discovery	57
Concepts for Discovery	57
NNMi Derives Attributes through Device Profiles	58
Plan Discovery	59
Select Your Primary Discovery Approach	59
List-Based Discovery	59
Rule-Based Discovery	59
Auto-Discovery Rules	60
Auto-Discovery Rule Ordering	60
Exclude Devices from Discovery	60
Ping Sweep	61
Discovery Seeds for Auto-Discovery Rules	61
Best Practices for Auto-Discovery Rules	61
Examples	61
Node Name Resolution	62
Subnet Connection Rules	62
Discovery Seeds	63
Rediscovery Interval	63
Do Not Discover Objects	64
Configure Discovery	64
Tips for Configuring Auto-Discovery Rules	64
Tips for Configuring Seeds	65
Evaluate Discovery	66
Follow the Progress of Initial Discovery	66
Were All Seeds Discovered?	66
Do All Nodes Have a Valid Device Profile?	67
Were All Nodes Discovered Properly?	67
Auto-Discovery Rules	67
IP Address Ranges	67
System Object ID Ranges	68

Are All Connections and VLANs Correct?	68
Rediscover a Device	68
Tune Discovery	68
NNMi State Polling	71
Concepts for State Polling	71
Plan State Polling	72
Polling Checklist	72
What Can NNMi Monitor?	73
Interfaces to Unmonitored Nodes	74
Stop Monitoring	75
Planning Groups	75
Interface Groups	76
Node Groups	76
Planning Polling Intervals	77
Deciding What Data to Collect	78
Configure State Polling	78
Configure Interface Groups and Node Groups	78
Configure Interface Monitoring	79
Configure Node Monitoring	79
Verify Default Settings	80
Evaluate State Polling	80
Verify the Configuration for Network Monitoring	80
Is the interface or node a member of the right group?	80
Which settings are being applied?	81
Which data is being collected?	81
Evaluate the Performance of Status Polling	81
Is the State Poller keeping up?	82
Tune State Polling	82
NNMi Incidents	85
Concepts for Incidents	85
Incident Lifecycle	86
Trap and Incident Forwarding	87
Comparison: Forwarding Third-Party SNMP Traps to Another Application	88
MIBs	89
Custom Incident Attributes	90
CIAs Added to Closed Management Event Incidents	90
Incident Reduction	91
Incident Suppression, Enrichment, and Dampening	92
Lifecycle Transition Actions	93
Plan Incidents	93
Which Device Traps Should NNMi Process?	93
Which Incidents Should NNMi Display?	94
How Should NNMi Respond to Incidents?	94
Should NNMi Receive Traps from an NNM Management Station?	94
Should NNMi Forward Traps to Another Event Receiver?	94

Configure Incidents	94
Configuring Incident Suppression, Enrichment, and Dampening	95
Configuring Lifecycle Transition Actions	95
Evaluate Incidents	95
Tune Incidents	96

Advanced Configuration 97

Working with Certificates for NNMi 99

Putting it All Together	100
Generating a Certificate Authority Certificate	101
Configuring Application Failover to use Self-Signed Certificates	103
Configuring Application Failover to use a Certificate Authority	104
Configuring High Availability to use Self-Signed or Certificate Authority Certificates	105
Configuring High Availability to use Self-Signed Certificates	105
Configuring High Availability for a New Certificate	105
Configuring the Global Network Management Feature to use Self-Signed Certificates	106
Configuring the Global Network Management Feature to use a Certificate Authority	107
Configuring Global Network Management with Application Failover to use Self-Signed Certificates	109
Configuring an SSL Connection to the Directory Service	110

Using Single Sign-On with NNMi 113

SSO Access for NNMi	113
SSO Access for NNMi and the iSPIs	114
Configuring Single Sign-On for the HP NNMi–HP BAC My BSM Integration	115
Disabling SSO	116
SSO Security Notes	116

Integrating NNMi with a Directory Service through LDAP 119

NNMi User Access Information and Configuration Options	119
Option 1: All NNMi User Information in the NNMi Database	120
Option 2: Some NNMi User Information in the NNMi Database and Some User Information in the Directory Service	121
Option 3: All NNMi User Information in the Directory Service	122
Configuring NNMi to Access a Directory Service	123
Directory Service Queries	128
Directory Service Access	128
Directory Service Content	129
Information Owned by the Directory Service Administrator	132
User Identification	133
Configuring NNMi User Access from the Directory Service (Detailed Approach)	134
Role Identification	135
Active Directory Role Identification	136
Other Directory Services Role Identification	136
Configuring NNMi Role Retrieval from the Directory Service (Detailed Approach)	137
Directory Service Configuration for Storing NNMi Roles	139
Troubleshooting the Directory Service Integration	139

ldap.properties Configuration File Reference	140
Examples	144
Global Network Management	145
Global Network Management Benefits	145
Is Global Network Management a Good Tool for Managing my Network?	146
Do I Need Continuous Multi-Site Network Monitoring?	146
Can my Critical Devices be Visible?	146
Licensing Considerations	146
Practical Global Network Management Examples	147
Review the Requirements	148
Regional Manager and Global Manager Connections	149
Initial Preparation	149
Port Availability: Configuring the Firewall	149
Configuring Self-Signed Certificates	149
Configuring Global Network Management for Application Failover	150
NNMi Management Server Sizing Considerations	150
Synchronizing System Clocks	150
Using the Application Failover Feature with Self-Signed Certificates in Global Network Management	150
Using Self-Signed Certificates in Global Network Management	151
Using a Certificate Authority in Global Network Management	151
List the Critical Equipment you Want to Monitor	151
Review the Global and Regional Managers' Management Domains	151
Review NNMi Help Topics	152
Configuring Single Sign-On for Global Network Management	152
Configuring Forwarding Filters on the Regional Managers	154
Configuring a Forwarding Filter to Limit Forwarded Nodes	154
Connecting a Global Manager with a Regional Manager	162
Determining the Connection States from global1 to regional1 and regional2	165
Reviewing global1 Inventory	166
Disconnecting Communication between global1 and regional1	167
Additional Information	170
Discovery and Data Synchronization	170
Status Poll or Configuration Poll a Device	172
Determining Device Status and NNMi Incident Generation using a Global Manager	173
Configuring Application Failover for Global Network Management	173
Configuring Application Failover on a Global Manager	174
Troubleshooting Tips for Global Network Management	175
Troubleshooting Information in the NNMi Help	175
Clock Synchronization	175
Global Network Management System Information	176
Synchronize Regional Manager Discovery from a Global Manager	176
Remedying a Destroyed Database on global1	176
Global Network Management and NNM iSPIs or Third-Party Integrations	177
Configuring NNMi for Application Failover	179
Application Failover Overview	179
Application Failover Basic Setup	180

Configuring NNMi for Application Failover	181
Using the Application Failover Feature	184
Application Failover Behavior Using the Embedded Database	185
Application Failover Behavior Using an Oracle Database	186
Application Failover Scenarios	186
Additional ovstart and ovstop Options	187
Application Failover Incidents	187
iSPIs and Application Failover	187
NNM iSPI Installation Information	188
Integrated Applications	189
Disabling Application Failover	190
Administrative Tasks and Application Failover	191
Application Failover and Upgrading to NNMi 9.00	192
Application Failover and NNMi Patches	193
Application Failover and Restarting the NNMi Management Servers	194
Application Failover and Recovery from a Previous Database Backup (Embedded Database Only)	195
Network Latency/Bandwidth Considerations	196
Application Failover and the NNMi Embedded Database	196
Network Traffic in and Application Failover Environment	197
An Application Failover Traffic Test	198
Configuring NNMi in a High Availability Cluster	201
Supported HA Products	202
Prerequisites to Configuring NNMi for HA	202
HA Concepts	203
HA Terms	204
NNMi HA Cluster Scenarios	205
Manpages	208
Configuring HA	209
Configuring NNMi Certificates for HA	209
Configuring NNMi for HA	209
NNMi HA Configuration Information	210
Configuring NNMi on the Primary Cluster Node	211
Configuring NNMi on the Secondary Cluster Nodes	214
Configuring NNM iSPIs for HA	215
NNM iSPI for Metrics, NNM iSPI for QA, and NNM iSPI for Traffic	215
NNM iSPI for MPLS, NNM iSPI for IP Multicast, and NNM iSPI for IP Telephony	215
NNM iSPI Network Engineering Toolset Software and NNMi Running under HA	215
Configuring NNMi for HA in an Oracle Environment	216
NNMi with Oracle HA Configuration Information	216
Configuring NNMi with Oracle for HA	216
Shared NNMi Data	218
Data on the NNMi Shared Disk	218
Replication of Configuration Files	219
Configuring the Shared Disk	219
A Note about Shared Disk Configuration on Windows Server	219
Licensing NNMi in an HA Cluster	220
Maintaining the HA Configuration	221

Maintenance Mode	221
Putting an HA Resource Group into Maintenance Mode	221
Removing an HA Resource Group from Maintenance Mode	221
Maintaining NNMi in an HA Cluster	221
Starting and Stopping NNMi	221
Changing NNMi Hostnames and IP Addresses in a Cluster Environment	221
Stopping NNMi Without Causing Failover	224
Restarting NNMi after Maintenance	224
Maintaining Add-on NNM iSPIs in an NNMi HA Cluster	224
Unconfiguring HA	225
Unconfiguring NNMi from an HA Cluster	225
Patching NNMi under HA	229
Upgrading NNMi under HA from NNMi 8.1x to NNMi 9.00	230
Troubleshooting the HA Configuration	234
General HA Troubleshooting	234
Error: Wrong Number of Arguments	234
Product Startup Times Out (Solaris)	234
Log Files on the Active Cluster Node Are Not Updating	234
Cannot Start the HA Resource Group on a Particular Cluster Node	235
NNMi-Specific HA Troubleshooting	236
Re-Enable NNMi for HA after All Cluster Nodes are Unconfigured	236
NNMi Does Not Start Correctly Under HA	236
nmsdbmgr Does Not Start after HA Configuration	237
pmd Does Not Start after HA Configuration	237
Disk Failover Does Not Occur	237
Shared Disk Files Are Not Found on the Secondary Node after Failover	237
NNM iSPI-Specific HA Troubleshooting	238
HA Configuration Reference	239
NNMi HA Configuration Files	239
NNMi-Provided HA Configuration Scripts	239
NNMi HA Configuration Log Files	241
Configuring NNMi Advanced for IPv6	243
Feature Description	243
Prerequisites	244
Licensing	245
Supported Configuration	245
Management Server	245
ISupported SNMP MIBs for IPv6	246
Installing NNMi	247
Activating IPv6 Features	247
Deactivating IPv6 Features	249
IPv6 Monitoring Following Deactivation	249
IPv6 Inventory Following Deactivation	249
Known Issues When Cleaning Up IPv6 Inventory	250

- NNMi Backup and Restore Tools 255**
 - Backup and Restore Commands 255
 - Backing up NNMi Data 256
 - Backup Type 256
 - Backup Scope 257
 - Restoring NNMi Data 259
 - Same System Restore 259
 - Different System Restore 260
 - Backup and Restore Strategies 260
 - Back up All Data Periodically 260
 - Back up Data Before Changing the Configuration 261
 - Back up Data Before Upgrading NNMi or the Operating System 261
 - Restore File System Files Only 261
 - Backing up and Restoring the Embedded Database Only 262

- Maintaining NNMi 263**
 - Administering a Custom Poller Collection Export 263
 - Changing the Custom Poller Collections Export Directory 263
 - Changing the Maximum Amount of Disk Space for Custom Poller Collections Export 264
 - Changing the Custom Poller Metric Accumulation Interval 264
 - Administering Incident Actions 265
 - Setting the Number of Simultaneous Actions 265
 - Setting the Number of Threads for Jython Actions 266
 - Setting the Action Server Name Parameter 266
 - Changing the Action Server Queue Size 267
 - Modifying NNMi Normalization Properties 267
 - Changing Normalization Properties Following an Initial Discovery 268
 - Configure HTTPS-Only Communication with the NNMi Console 269
 - NNMi Self Monitoring 269

- Configuring the Telnet and SSH Protocols for Use by NNMi 271**
 - Disable the Telnet Menu Item 271
 - Configure a New Menu Item that uses the ssh Protocol 272
 - Configure a Telnet or SSH Client for the Browser on Windows 272
 - Windows Operating System-Provided Telnet Client 274
 - Third-Party Telnet Client (Standard Windows) 275
 - Third-Party Telnet Client (Windows on Windows) 276
 - Third-Party SSH Client (Standard Windows and Windows on Windows) 277
 - Configure Firefox to use Telnet or SSH on Linux 279
 - Telnet on Linux 279
 - Secure Shell on Linux 280
 - Example Files for Changing the Windows Registry 280
 - Example nnmtnet.reg 281
 - Example nnmputtytelnet.reg 281
 - Example nnmtnet32on64.reg 281
 - Example nnmssh.reg 281

Changing the NNMi Management Server	283
Best Practices for Preparing the NNMi Configuration to Be Moved	283
Moving the NNMi Configuration and Embedded Database	284
Moving the NNMi Configuration	284
Restoring the NNMi Public Key Certificate	285
Changing the IP Address of an NNMi Management Server	288
Changing the Hostname or Domain Name of an NNMi Management Server	289
Changing the Oracle Database Instance Connection Information	292
Changing the Password that NNMi Uses to Connect to the Oracle Database Instance	293
Upgrading to NNMi 9.00	295
Upgrading the NNMi Management Server in Place	297
Start from NNMi 8.0x	297
Upgrade an Existing NNMi Management Server to NNMi 9.0x Patch 1	297
Upgrading to a Different NNMi Management Server	299
Start from NNMi 8.0x	299
Upgrade to a Different NNMi Management Server	299
Moving NNMi from Red Hat Linux 4.6 to 5.2 or 5.3	301
Changing NNMi from Red Hat Linux 4.6 to Red Hat Linux 5.2 or 5.3	301
Migrating NNMi Oracle Data	305
Migrating NNMi Oracle Data	305
Additional Upgrade Information	307
Configuration Differences	307
Functionality Differences	308
Upgrading from NNM 6.x/7.x	311
Product Comparison	313
Network Discovery	313
Key Concepts for Discovery	315
Status Monitoring	316
Key Concepts for Status Monitoring	318
Customizing Event Monitoring	318
Key Concepts for Event Monitoring	320
Upgrading from NNM 6.x/7.x	321
Upgrade Options	322
A Fresh Beginning	322
Upgrading in Phases	322
Phase 1: Collect Data from the NNM Management Station	325
Phase 2: Upgrade SNMP Information	327
Configure SNMP Access	327
Limit Name Resolution	331
Customize Device Profiles	332

Phase 3: Upgrade Discovery	333
Schedule Discovery	334
Select Your Discovery Method	335
Configure Auto-Discovery Rules	336
Configure Spiral Discovery	336
Exclude Addresses from Discovery	340
Add Seeds to NNMi for Seeded Discovery	341
Customize Connectivity	342
Phase 4: Upgrade Status Monitoring	343
Set Polling Intervals	343
Select Polling Protocol	345
Configure Critical Nodes	347
Exclude Objects from Status Polling	348
Phase 5: Upgrade Event Configuration and Event Reduction	349
Display Traps from Devices	349
Customize Display of NNMi-Generated Management Events	351
Block/Ignore/Disable Traps	351
Configure Lifecycle Transition Actions	352
Configure Additional (Manual) Actions	353
Event Correlation: Repeating Events	354
Event Correlation: Counting the Rate	355
Event Correlation: Pairwise Cancellation	355
Event Correlation: Scheduled Maintenance	355
Phase 6: Upgrade Graphical Visualization (OVW)	356
Phase 6: Upgrade Graphical Visualization (Home Base)	358
Phase 7: Upgrade Custom Scripts	359
Upgrade Tools Reference	360
Data Collection Tools	360
NNM Configuration Data Files	362
Data Import Tools for Upgrading	363
Integrating NNM 6.x or NNM 7.x with NNMi	365
Configure Event Forwarding	366
Step 1: Configure NNM 6.x/7.x to Forward Events to the NNMi Management Server	366
Recommended and Supported Procedure: Use the Event Configuration Window	366
Optional: Destination List File	367
Alternative Procedure: Manually Edit trapd.conf	368
Step 2: (Optional) Use Node Level Filtering to Further Reduce Events	368
Step 3: Add the NNM 6.x/7.x Management Station to the NNMi Topology	368
Step 4: (Optional) Save the Management Station Configuration	369
Step 5: Verify NNM 6.x/7.x Incident Configuration in the NNMi Console	369
Mapping Categories	369
Configure Remote View Launching	370
Step 1: Install Java Plug-in	370
Step 2: Create an NNM 6.x/7.x Management Station Entity in NNMi	371
Step 3: (Optional) Configure Additional NNM 6.x/7.x Views	372
URLs That Do Not Require a Selection	372
URLs That Require a Selection	373

Test the Integration	373
Test 1: Verify Event Forwarding	373
Generate Test Interface Down and Interface Up Events	374
sendMsg.ovpl	375
Test with Traps to NNM 6.x/7.x System	375
Test 2: Launch NNM 6.x/7.x Dynamic Views from NNMi	376
Troubleshoot Event Forwarding	376

Integrations with NNMi 379

AlarmPoint 381

HP NNMi–AlarmPoint Integration	382
About the HP NNMi–AlarmPoint Integration	382
Value	382
Supported Versions	383
Documentation	383
Enabling the HP NNMi–AlarmPoint Integration	383
Using the HP NNMi–AlarmPoint Integration	384
Disabling the HP NNMi–AlarmPoint Integration	385
Troubleshooting the HP NNMi–AlarmPoint Integration	385
HP NNMi–AlarmPoint Mobile Gateway Integration	385
About the HP NNMi–AlarmPoint Mobile Gateway Integration	385
Value	385
Supported Versions	385
Documentation	386
Enabling the HP NNMi–AlarmPoint Mobile Gateway Integration	386
Using the HP NNMi–AlarmPoint Mobile Gateway Integration	386
Disabling the HP NNMi–AlarmPoint Mobile Gateway Integration	387
Troubleshooting the HP NNMi–AlarmPoint Mobile Gateway Integration	387

CiscoWorks LAN Management Solution 389

HP NNMi–CiscoWorks LMS Integration	389
Value	389
Supported Versions	390
Documentation	390
Enabling the HP NNMi–CiscoWorks LMS Integration	390
Using the HP NNMi–CiscoWorks LMS Integration	391
Changing the HP NNMi–CiscoWorks LMS Integration Configuration	392
Disabling the HP NNMi–CiscoWorks LMS Integration	392
Troubleshooting the HP NNMi–CiscoWorks LMS Integration	392
CiscoWorks LMS Actions Do Not Work	392
OID Not Found in the MIB Cache Message in Traps	392
HP NNMi–CiscoWorks LMS Integration Configuration Form Reference	393
NNMi Management Server Connection	393
CiscoWorks LMS Server Connection	394

Clarus Systems ClarusIPC Plus+	395
HP NNMi–Clarus Systems ClarusIPC Plus+ Integration	395
About the HP NNMi–Clarus Systems ClarusIPC Plus+ Integration	395
Value	396
Supported Versions	396
Documentation	396
Enabling the HP NNMi–Clarus Systems ClarusIPC Plus+ Integration	396
Using the HP NNMi–Clarus Systems ClarusIPC Plus+ Integration	396
Disabling the HP NNMi–Clarus Systems ClarusIPC Plus+ Integration	396
Troubleshooting the HP NNMi–Clarus Systems ClarusIPC Plus+ Integration	397
HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus+ Integration	397
About the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus+ Integration	397
Value	397
Supported Versions	398
Documentation	398
Enabling the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus+ Integration	398
Using the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus+ Integration	399
Disabling the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus+ Integration	399
Troubleshooting the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus+ Integration	399
HP Asset Manager	401
HP NNMi–HP Asset Manager Integration	401
Value	401
Supported Versions	402
Documentation	402
Using the HP NNMi–HP Asset Manager Integration	402
HP Business Availability Center My BSM	403
HP NNMi–HP BAC My BSM Integration	403
Value	404
Supported Versions	404
Documentation	404
Default NNMi Modules for My BSM	404
Configuring the Demonstration Portlets	406
Creating Custom NNMi Portlets	407
Determining the Portlet URL	407
Portlet Definition HTML Reference	408
Configuring Single Sign-On for the HP NNMi–HP BAC My BSM Integration	410
Troubleshooting the HP NNMi–HP BAC My BSM Integration	410
The NNMi Portlets Appear As a Sign-in Page	410
An NNMi Portlet Does Not Load Correctly	411
An NNM iSPI for Performance Portlet Does Not Load Correctly	411
An NNM iSPI for Performance Portlet Displays an AsynchWait_Requests Error	411
Single Sign-On Does Not Work Correctly	411
My BSM Reports HTML Validation Errors When I Save A Portlet Definition	411

HP NNMi—HP BAC My BSM Configuration Form Reference	412
HP Business Service Management Topology	415
HP NNMi—HP BSM Topology Integration	415
Value	416
Supported Versions	416
Documentation	416
Enabling the HP NNMi—HP BSM Topology Integration	416
Using the HP NNMi—HP BSM Topology Integration	417
Changing the HP NNMi—HP BSM Topology Integration Configuration	418
Disabling the HP NNMi—HP BSM Topology Integration	418
Troubleshooting the HP NNMi—HP BSM Topology Integration	418
HP NNMi—HP BSM Topology Integration Configuration Form Reference	419
NNMi Management Server Connection	419
BSM Gateway Server Connection	420
BSM Topology Filter	421
HP Universal CMDB	423
HP NNMi—HP UCMDB Integration	423
Value	424
Supported Versions	424
Documentation	424
Using the HP NNMi—HP UCMDB Integration	424
HP Network Automation	425
HP NNMi—HP NA Integration	425
Value	426
Supported Versions	426
Documentation	427
Enabling the HP NNMi—HP NA Integration	427
Using the HP NNMi—HP NA Integration	429
NNMi Functionality Provided by the Integration	429
Configuring NA Diagnostics and Command Scripts as Incident Actions	430
Viewing the Results of Incident Actions that Access NA	431
Identifying Layer 2 Connections with Mismatched States (NNM iSPI NET)	431
NA Functionality Provided by the Integration	432
Disabling Network Management During Device Configuration	432
Propagating Device Community String Changes	433
Importing NNMi 8.10 Devices into the NA Inventory	433
Changing the HP NNMi—HP NA Integration	434
Disabling the HP NNMi—HP NA Integration	434
Troubleshooting the HP NNMi—HP NA Integration	434
HP NNMi—HP NA Integration Configuration Form Reference	437
NNMi Management Server Connection	437
NA Server Connection	438
Integration Behavior	438

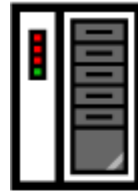
HP ProCurve Manager Plus	439
HP NNMi–HP ProCurve Manager Plus Integration	439
Value	440
Supported Versions	440
Documentation	440
Using the HP NNMi–HP ProCurve Manager Plus Integration	440
HP RAMS MPLS WAN	441
HP NNMi–HP RAMS MPLS WAN Integration	441
Value	441
Supported Versions	441
Documentation	442
Using the HP NNMi–HP RAMS MPLS WAN Integration	442
HP Systems Insight Manager	443
HP NNMi–HP SIM Integration	444
Value	444
Supported Versions	444
Documentation	444
Enabling the HP NNMi–HP SIM Integration	444
Using the HP NNMi–HP SIM Integration	446
Changing the HP NNMi–HP SIM Integration Configuration	446
Disabling the HP NNMi–HP SIM Integration	446
Troubleshooting the HP NNMi–HP SIM Integration	447
SIM Actions Do Not Work	447
OID Not Found in the MIB Cache Message in Traps	447
HP NNMi–HP SIM Integration Configuration Form Reference	447
NNMi Management Server Connection	448
SIM Server Connection	448
nGenius Performance Manager	451
HP NNMi–nGenius Performance Manager Integration	452
Value	452
Supported Versions	453
Documentation	453
Enabling the HP NNMi–nGenius Performance Manager Integration	453
Using the HP NNMi–nGenius Performance Manager Integration	454
Disabling the HP NNMi–nGenius Performance Manager Integration	454
Troubleshooting the HP NNMi–nGenius Performance Manager Integration	454
NNMi Northbound Interface	455
NNMi Northbound Interface	456
Value	456
Supported Versions	456
Terminology	456
Documentation	457
Enabling the NNMi Northbound Interface	457
NNMi Northbound Interface Upgraded from NNMi 8.1x	457
New NNMi Northbound Interface Configuration	458

Using the NNMi Northbound Interface	459
Incident Forwarding	459
Incident Lifecycle State Change Notifications	460
Incident Correlation Notifications	461
Incident Deletion Notifications	462
Event Forwarding Filter	462
Changing the NNMi Northbound Interface	463
Disabling the NNMi Northbound Interface	463
Troubleshooting the NNMi Northbound Interface.	463
Application Failover and the NNMi Northbound Interface.	465
Local Northbound Application	465
Remote Northbound Application	465
NNMi Northbound Interface Destination Form Reference	466
Northbound Application Connection Parameters	466
NNMi Northbound Interface Integration Content	467
NNMi Northbound Interface Destination Status Information	469
HP BSM Operations Management	471
HP NNMi—HP BSM Operations Management Integration	472
Value	472
Supported Versions	473
Documentation	473
Enabling the HP NNMi—HP BSM Operations Management Integration.	473
Using the HP NNMi—HP BSM Operations Management Integration	476
Configuration Item Identifiers	476
Health Indicators	477
Default Policy Conditions	477
Customizing Policy Conditions	478
More Information	478
Changing the HP NNMi—HP BSM Operations Management Integration	479
Update the SNMP Trap Policy Conditions for New NNMi Traps	479
Change the Configuration Parameters	479
Disabling the HP NNMi—HP BSM Operations Management Integration	480
Troubleshooting the HP NNMi—HP BSM Operations Management Integration.	480
BSM Operations Management Event Browser Contains No Forwarded Incidents	480
BSM Operations Management Event Browser Contains Only Some Forwarded Incidents.	482
HP NNMi—HPOM Agent Destination Form Reference (BSM Operations Management Integration)	483
BSM Integration Adapter Connection	483
BSM Operations Management Integration Content	484
BSM Integration Adapter Destination Status Information	486
HP Operations Manager	487
HP NNMi—HPOM Integration (Agent Implementation)	487
About the HP NNMi—HPOM Integration (Agent Implementation).	488
Value	488
Supported Versions	489
Documentation	489

Enabling the HP NNMi–HPOM Integration (Agent Implementation)	490
Integration Configuration Upgraded from NNMi 8.1x	490
New Integration Configuration	490
Using the HP NNMi–HPOM Integration (Agent Implementation)	494
Configuration Item Identifiers	494
Health Indicators	495
Default Policy Conditions	495
Customizing Policy Conditions	496
More Information	496
Changing the HP NNMi–HPOM Integration Configuration (Agent Implementation)	497
Update the SNMP Trap Policy Conditions for New NNMi Traps	497
Change the Configuration Parameters	497
Disabling the HP NNMi–HPOM Integration (Agent Implementation)	498
Troubleshooting the HP NNMi–HPOM Integration (Agent Implementation)	498
HPOM Active Messages Browser Does Not Receive Any Forwarded Incidents	498
HPOM Active Messages Browser Does Not Receive Some Forwarded Incidents	500
HP NNMi–HPOM Agent Destination Form Reference (Agent Implementation)	502
HP Operations Agent Connection	502
HPOM Integration Content	503
HP Operations Agent Destination Status Information	505
HP NNMi–HPOM Integration (Web Services Implementation)	506
About the HP NNMi–HPOM Integration (Web Services Implementation)	506
Value	507
Supported Versions	508
Documentation	508
Enabling the HP NNMi–HPOM Integration (Web Services Implementation)	508
HPOM for Windows	508
HPOM for UNIX and HPOM for Linux	510
Using the HP NNMi–HPOM Integration (Web Services Implementation)	512
Usage Example	512
A Normal Situation: Unknown MSI Condition	513
More Information	513
Changing the HP NNMi–HPOM Integration Configuration (Web Services Implementation)	513
Disabling the HP NNMi–HPOM Integration (Web Services Implementation)	514
For All HPOM Management Servers	514
For One HPOM Management Server	514
Troubleshooting the HP NNMi–HPOM Integration (Web Services Implementation)	514
HPOM Does Not Receive Any Forwarded Incidents	514
HPOM Does Not Receive Some Forwarded Incidents	517
NNMi Incident Information Is Not Available in the HPOM Messages Browser	517
NNMi and HPOM Are Not Synchronized	517
The Integration Does Not Work Through a Firewall	518

HP NNMi–HPOM Web Services Integration Configuration Form Reference	519
NNMi Management Server Connection	519
HPOM Management Server Connection	520
Integration Behavior	521
Incident Filters	522
Example Incident Filters.	523
Incident Filter Limitations	524
HP NNMi Integration Module for Netcool Software	525
HP NNMi Integration Module for Netcool Software	525
Value.	526
Supported Versions	526
Documentation.	527
Enabling the HP NNMi Integration Module for Netcool Software	527
NNMi Integration Module for Netcool Software Upgraded from NNMi 8.1x	527
New NNMi Integration Module for Netcool Software Configuration	528
Using the HP NNMi Integration Module for Netcool Software	530
Changing the HP NNMi Integration Module for Netcool Software	531
Disabling the HP NNMi Integration Module for Netcool Software	531
Troubleshooting the HP NNMi Integration Module for Netcool Software	532
Netcool/OMNIBus Does Not Receive Any Forwarded NNMi Management Events	532
Netcool/OMNIBus Does Not Receive Some Forwarded NNMi Management Events	533
Error When Launching an NNMi Form for a Layer 2 Connection	533
HP NNMi Integration Module for Netcool Software Destination Form Reference	534
Netcool/OMNIBus SNMP Probe Connection	534
Integration Content	535
Destination Status Information	538
Additional Information	539
NNMi Environment Variables	541
Environment Variables Used in This Document	541
Other Available Environment Variables.	542
Windows Paths and Environment Variables from NNMi 8.00	545
NNMi 9.00 and Well-Known Ports	547
Suggested Configuration Changes	551
Problems and Solutions	553
Glossary	555
Index	563

About This Guide



(1) First installation
or test bed

Follow steps in
NNMi Installation Guide



(2) Production deployment and
migration from previous versions

Read NNMi Deployment
Reference (this book)



This chapter contains the following topics:

- [What Is in This Guide?](#)
- [Environment Variables Used in This Document](#)
- [Revision History](#)
- [For More Information about NNMi](#)

What Is in This Guide?

This guide contains a collection of information and best practices for deploying HP Network Node Manager i Software, including NNMi and NNMi Advanced. This guide is for an expert system administrator, network engineer, or HP support engineer with experience deploying and managing networks in large installations.

This guide assumes that you have already installed NNMi in a limited (test) environment, and that you are familiar with start-up configuration tasks, such as using the Quick Start Configuration wizard to configure community strings, set up discovery for a limited range of network nodes, and create an initial administrator account. To learn more about these tasks, see the *NNMi Installation Guide* (see [Available Product Documentation](#) on page 3).

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.

Environment Variables Used in This Document

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

- *Windows Server 2003:*

- %NnmInstallDir%: <drive>\Program Files\HP\HP BTO Software
- %NnmDataDir%: <drive>\Documents and Settings\All Users\Application Data\HP\HP BTO Software



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



If you first installed NNMi 8.00, your system uses different values for these environment variables, as described in [Windows Paths and Environment Variables from NNMi 8.00](#) on page 545.

- *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](#) on page 542.

Revision History

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

Document Release Date	Description of Major Changes
NNMi version 9.0x:	
April 2010 (9.00)	Entirely updated. <ul style="list-style-type: none">• Third English edition.• Second Japanese edition.
June 2010 (patch 1)	All changes are English only. <ul style="list-style-type: none">• Updated NNMi Incidents:<ul style="list-style-type: none">— Added CIAs Added to Closed Management Event Incidents.— Added Lifecycle Transition Actions.— Added Configuring Incident Suppression, Enrichment, and Dampening.— Added Configuring Lifecycle Transition Actions.• Updated Configuring NNMi for Application Failover with additional information in Network Latency/Bandwidth Considerations.• Updated AlarmPoint to reflect that the NNMi integration enablement license is <i>not</i> required for NNMi integrations with any of the AlarmPoint products.• Updated HP Business Service Management Topology:<ul style="list-style-type: none">— Updated HP NNMi–HP BSM Topology Integration.— Updated Using the HP NNMi–HP BSM Topology Integration.• Added HP BSM Operations Management.

For More Information about NNMi

To obtain a complete set of information about the NNMi product, use this guide along with other NNMi documentation. The table below shows all NNMi documents to date, including both guides and white papers.



All information below can be downloaded from <http://h20230.www2.hp.com/selfsolve/manuals>. See [Available Product Documentation](#) on page 3 for more information.

What do you want to do?	Where to find more information
View a list of available documentation for this version of NNMi.	Download the <i>NNMi Documentation List</i> . 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.
Install NNMi or NNMi Advanced (first time).	Download the <i>NNMi Installation Guide</i> . This guide contains basic steps to install and un-install the product, plus how to do an initial configuration using the NNMi Quick Start Configuration Wizard. <ul style="list-style-type: none"> • <i>HP Network Node Manager i Software Installation Guide for the Windows Operating System</i> • <i>HP Network Node Manager i Software Installation Guide for the HP-UX Operating System</i> • <i>HP Network Node Manager i Software Installation Guide for the Linux Operating System</i> • <i>HP Network Node Manager i Software Installation Guide for the Solaris Operating System</i>
Plan for network deployment, including links to system requirements.	See Preparation on page 29 of this guide.
Configure NNMi for a production environment.	See Configuration on page 35 of this guide.
Configure NNMi behind the scenes.	See Advanced Configuration on page 97 of this guide.
Maintain the NNMi configuration.	See Maintaining NNMi on page 253 of this guide.
Upgrade to NNMi from previous versions of Network Node Manager i Software.	See Upgrading to NNMi 9.00 on page 295 of this guide.
Upgrade to NNMi from previous versions of Network Node Manager.	See Upgrading from NNM 6.x/7.x on page 311 of this guide.
Learn more about products that integrate with NNMi.	See Integrations with NNMi on page 379 of this guide.
Reference NNMi environment variables, ports, and messages.	See Additional Information on page 539 of this guide.
Obtain more information about a specific topic.	Download by example documents and white papers.

What do you want to do?	Where to find more information
Print the NNMi help.	Download PDFs of the help content.
Install the HP NNM iSPI Network Engineering Toolset (NNM iSPI NET) diagnostics server and learn about NNM iSPI NET functionality.	Download the <i>HP NNM iSPI Network Engineering Toolset Planning and Installation Guide</i> from the Network Node Manager SPI for NET product category for the Windows operating system.
Obtain documentation for the NNM Developer's Toolkit (SDK).	Download the <i>HP Network Node Manager i Software Developer's Toolkit Guide</i> .

Preparation

This section contains the following chapter:

- [Hardware and Software Requirements](#)

Hardware and Software Requirements

This chapter contains the following topics:

- Supported Hardware and Software
- System Configuration (UNIX)
- NNMi Coexistence with HP Performance Insight
- NNMi Coexistence with HP Operations Agent

Supported Hardware and Software

Before installing NNMi, read the information about NNMi hardware and software requirements described in [Table 1](#).



For current versions of all documents listed here, go to:

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

Table 1 Software and Hardware Pre-Installation Checklist

Complete (y/n)	Document to Read
	<i>NNMi Installation Guide</i> <ul style="list-style-type: none"> • Filename = install-guide_en.pdf • Windows Media = DVD main drive (root) • UNIX Media = Root directory • NNMi console = Help > NNMi Documentation Library > Installation Guide
	<i>NNMi Release Notes</i> <ul style="list-style-type: none"> • Filename = releasenotes_en.html • Windows Media = DVD main drive (root) • UNIX Media = Root directory • NNMi console = Help > NNMi Documentation Library > Release Notes
	<i>NNMi System and Device Support Matrix</i> <ul style="list-style-type: none"> • Filename = supportmatrix_en.html • Windows Media = DVD main drive (root) • UNIX Media = Root directory • NNMi console = Linked from the release notes

➤ HP updates the *NNMi System and Device Support Matrix* as new information becomes available. Before you deploy NNMi, check for the most recent NNMi support matrix for your version of the software at:

http://www.hp.com/go/hpsoftwaresupport/support_matrices

(You must have an HP Passport ID to access this web site.)

➤ If you plan to install NNM Smart Plug-ins (NNM iSPIs), include the system requirements for those products as you plan the NNMi deployment.

System Configuration (UNIX)

If you cannot display NNMi manpages on the NNMi management server, verify that the MANPATH variable contains the /opt/OV/man location. If it does not, add the /opt/OV/man location to the MANPATH variable.

NNMi Coexistence with HP Performance Insight

If you plan to install NNMi on the same server as HP Performance Insight, follow this procedure to avoid problems with the installation sequence and port conflicts:

- 1 Install HP Performance Insight first.



Do not install NNMi until after you complete [step 1](#) and [step 2](#).

- 2 Stop all HP Performance Insight processes.
- 3 Install NNMi. See the *NNMi Installation Guide* for specific instructions.
- 4 Stop all NNMi processes:

```
ovstop -c
```
- 5 Modify the `nms-local.properties` file to resolve any port conflicts. You can find this file in the following directory:
 - *Windows*: %NNM_CONF%\nmm\props
 - *UNIX*: \$NNM_CONF/nmm/props
- 6 Start HP Performance Insight processes.
- 7 Start all NNMi processes:

```
ovstart -c
```



When NNMi is installed on the same server as HP Performance Insight, uninstalling NNMi causes an exception when running the HP PI MIB Browser. To prevent this exception, complete the following steps:

- 1 Uninstall NNMi.
- 2 Recreate the snmpmib MIB database:
 - a `mkdir -p /var/opt/OV/shared/nmm/conf/`
 - b `/opt/OV/sbin/nmmloadmib -load /usr/OVPI/mibs/GENMIB2IF.mib`
- 3 Use the `nmmloadmib.ovpl` command to load additional MIBs.

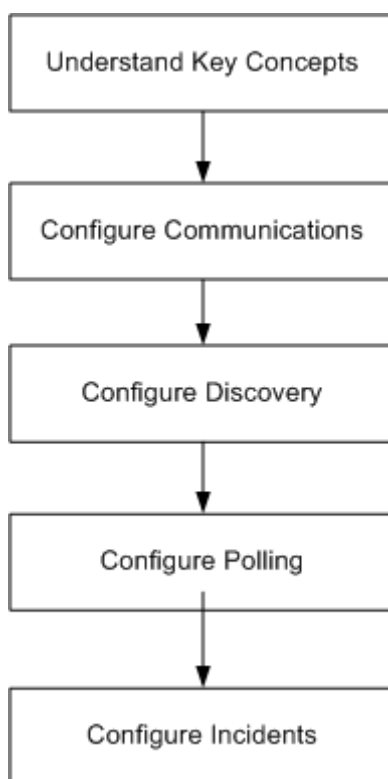
NNMi Coexistence with HP Operations Agent

If you plan to install an HP Operations agent on the NNMi management server (for communicating with HP Operations Manager (HPOM)), install NNMi before installing the HP Operations agent.

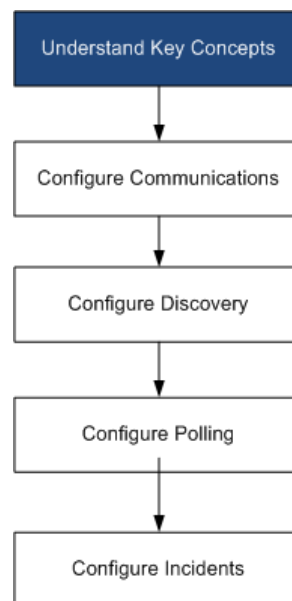
Configuration

This section contains the following chapters:

- [General Concepts for Configuration](#)
- [NNMi Communications](#)
- [NNMi Discovery](#)
- [NNMi State Polling](#)
- [NNMi Incidents](#)



General Concepts for Configuration



Read this chapter for an introduction to concepts that are explained in more detail later in this guide. This chapter also contains some best practices that apply to all HP Network Node Manager i Software configuration areas.

This chapter contains the following topics:

- [Task Flow Model](#)
- [Best Practice: Save the Existing Configuration](#)
- [Best Practice: Use the Author Attribute](#)
- [User Interface Model](#)
- [Ordering](#)
- [Node Groups](#)
- [Node/Interface/Address Hierarchy](#)
- [Stop Everything and Start Over Again](#)

Task Flow Model

The chapters in the configuration section of this guide support the following task flow:

- 1 **Concepts**—Gain a general understanding of the configuration area. The information in this guide supplements the information in the NNMi help.
- 2 **Plan**—Decide how you want to approach the configuration. This is a good time to begin or update your company's network management documentation.
- 3 **Configure**—Use a combination of the NNMi console, configuration files, and command line interface to enter the configuration into NNMi. See the NNMi help for specific procedures.
- 4 **Evaluate**—In the NNMi console, examine the results of your configuration. Adjust the configuration as necessary to achieve the desired results.
- 5 **Tune**—Optional. Adjust the configuration to improve NNMi performance.

Best Practice: Save the Existing Configuration

It is a good idea to save a copy of the existing configuration before you make any major configuration changes. If you do not like the results of your configuration changes, it is easy to revert to your saved configuration.

Use the `nnmconfigexport.ovpl` command to save the current configuration. To recover a saved configuration, use the `nnmconfigimport.ovpl` command.

For information about how to use 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.

Best Practice: Use the Author Attribute

Many NNMi configuration forms include the **Author** attribute.

As you create or modify the configurations on these forms, set the **Author** attribute to a value that identifies your organization. When you export the NNMi configuration, you can specify an author value to pull only those items that your organization has customized.

When you upgrade NNMi, the installer does not overwrite any configurations whose author value is not HP.

User Interface Model

Some NNMi console forms use a transactional approach to updating the database. The changes that you make in the NNMi console forms do not take effect until you save and close the forms all of the way back to the NNMi console. If you close a form that contains unsaved changes (on that form or on a contained form), NNMi warns you about the unsaved changes and gives you a chance to cancel the close.



The **Discovery Seed** form is one exception to the transactional approach. This form is provided on the **Discovery Configuration** form as a convenience, but it is disconnected from the rest of discovery configuration. For this reason, you must save and close the **Discovery Configuration** form to implement your auto-discovery *rules* before you configure any discovery seeds for those rules.

Ordering

Some NNMi console configuration forms include the **Ordering** attribute, which sets the priority for applying the configurations. For one configuration area, NNMi evaluates each item against the configurations from the smallest (lowest) ordering number to the next lowest ordering number, and so on, until NNMi finds a match. At that point,

NNMi uses the information from the matching configuration and ceases to look for any more matches. (The communication configuration is an exception. NNMi continues to search for information at other levels to complete the communication settings.)

The **Ordering** attribute plays an important role in NNMi configuration. If you see unexpected discovery or status results, check the ordering of the configurations for that area.

Ordering applies within the local context. The Menus and Menu Items tables contain multiple objects with the same ordering number because of the local context idea.

Ordering numbers are also used in the following places, but with different meanings:

- Ordering on the **Menu** and **Menu Item** forms sets the order of items in the local context of the associated menu.
- Topology maps ordering on the **Node Group Map Settings** form sets the order of items in the **Topology Maps** workspace.

For specific information about how the **Ordering** attribute affects a given configuration area, see the NNMi help for that area.

Best practice For each configuration area, apply low ordering numbers to the most restrictive configurations, and apply high ordering numbers to the least restrictive configurations.

Best practice For each configuration area, all ordering numbers must be unique. During initial configuration use ordering numbers with a standard interval to provide flexibility for future modifications to the configuration. For example, give the first three configurations the ordering numbers 100, 200, and 300.

Node Groups

In NNMi, the primary filtering technique is to group nodes or interfaces, and then applying settings to a group or filtering visualizations by group. Node groups can be used for any or all of the following purposes:

- Monitoring settings
- Incident payload filtering
- Table filtering
- Customizing map views
- Filtering the nodes passed from a regional manager to the global manager for the global network management feature

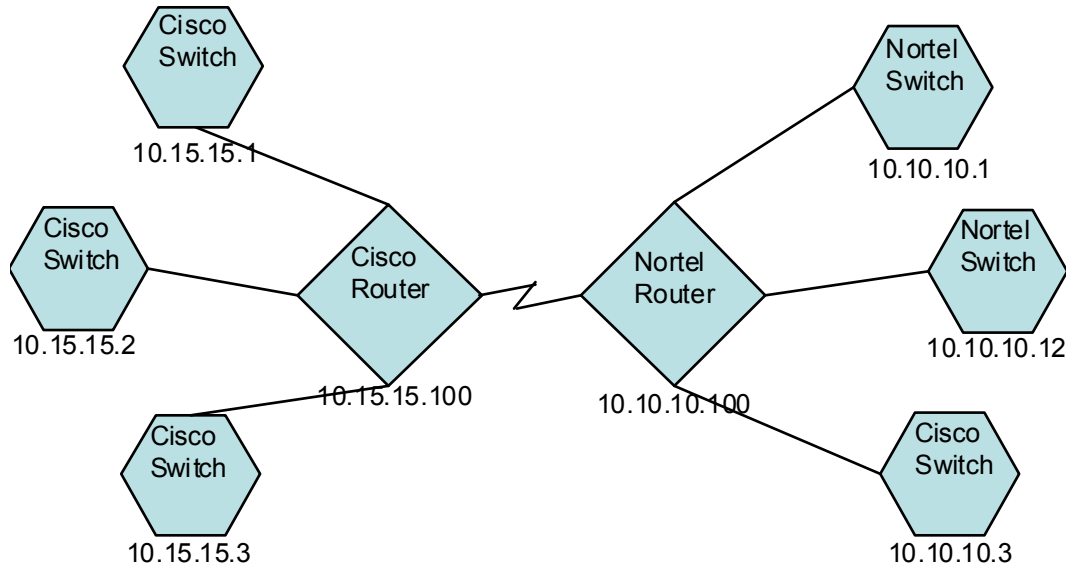
Interface groups can be used for either or both of the following purposes:

- Excluding interfaces from discovery
- Monitoring settings
- Incident payload filtering
- Table filtering

You can create a hierarchy of node groups based on any filterable attributes to control map view drill-down and/or monitoring setting inheritance.

Group Overlap

Regardless of the intended uses for group definitions, the first step is to define which nodes or interfaces are members of a group. Because you can create groups for different purposes, each object can be included in multiple groups. Consider the following example:



- For monitoring purposes, you might want to set a polling interval of 3 minutes for all switches, regardless of vendor or location. You can do this with a device category filter.
- For maintenance purposes, you might want to group all Cisco switches so that you can place them OUT OF SERVICE together for IOS upgrades. You can do this with a vendor filter.
- For visualization, you might want to group all devices on the 10.10.*.* site into a container with propagated status. You can do this with an IP address filter.

The Cisco switch with IP address 10.10.10.3 would qualify for all three groups.

You want to find the balance between having a useably rich set of groups available for configuration and viewing, and overloading the list with superfluous entries that will never be used.

Node Group Membership

NNMi determines node group membership by comparing each discovered node to each of the configured node groups.

- All nodes specified on the **Additional Nodes** tab are members of the node group.
- All nodes that are members of at least one node group specified on the **Child Node Groups** tab are members of the node group.
- Any node that matches one or more entries (if any exist) on the **Device Filters** tab *and* the filter specified on the **Additional Filters** tab is a member of the node group.

Hierarchies/Containment

You can create simple, reusable, atomic groups and combine them hierarchically for monitoring or visualization. Using hierarchical containers for nodes greatly enhances map views by providing cues about the location or type of object at fault. NNMi gives you complete control of the definition of the groups and their drill-down order.

You can create simple, reusable atomic groups first, and then specify them as child groups as you build up. Alternatively, you can specify your largest parent group first and create child groups as you go.

For example, a network might contain Cisco switches, Cisco routers, Nortel switches, and Nortel routers. You can create parent groups for Cisco devices and for all switches. Because the hierarchy is specified when you create the parent and designate its children, each child group, such as Cisco switches, can have multiple parents.

Hierarchies work well for the following situations:

- Types of nodes with similar monitoring needs
- Geographical locations of nodes
- Types of nodes to be taken OUT OF SERVICE together
- Groups of nodes by operator job responsibility

When you use groups in map views and table views, you see a (configurable) propagated status for the group.



Keep in mind that as you use group definitions to specify monitoring configuration, hierarchy does *not* imply ordering for settings. The settings with the lowest ordering number apply to a node. By carefully incrementing ordering numbers, you can emulate inheritance concepts for settings.

The configuration interface automatically prevents circular hierarchy definitions.

Device Filters

During discovery, NNMi collects direct information through SNMP queries and derives other information from that through device profiles. (For more information, see [NNMi Derives Attributes through Device Profiles](#) on page 58.) By gathering the system object ID, NNMi can index through the correct device profile to derive the following information:

- Vendor
- Device category
- Device family within the category

These derived values, in addition to the device profile itself, are available for use as filters.

For example, you can group all objects from a specific vendor, regardless of device type and family. Or you can group all devices of a type such as router, across vendors.

Additional Filters

With the additional filters editor, you can create custom logic to match fields including:

- hostname (Hostname)
- mgmtIPAddress (Management Address)
- hostedIPAddress (Address)
- sysName (System Name)
- sysLocation (System Location)
- sysContact (System Contact)
- capability (Unique Key of the Capability)
- customAttrName (Custom Attribute Name)
- customAttrValue (Custom Attribute Value)

Filters can include the AND, OR, NOT, EXISTS, NOT EXISTS, and grouping (parentheses) operations. For more information, see *Specify Node Group Additional Filters* in the NNMi help.

Capabilities are primarily intended for other programs that integrate with NNMi. For example, router redundancy and component health add capabilities (fields) to the NNMi database. You can view these capabilities by examining the node details from a device that has already been discovered.

Custom attributes can be added by iSPIs, or you can create your own custom attributes. If you have not purchased the Web Services SDK, you must place values in the field for each node manually. For example, an asset number or serial number might be an attribute that is not a capability.

Additional Nodes

If the network contains critical devices that are too difficult to qualify using filters, you can add them to a group by individual hostname.

Node Group Status

When configured to do so, NNMi determines the status of a node group using one of the following algorithms:

- Set the node group status to match the most severe status of any node in the node group. To use this approach, select the **Propagate Most Severe Status** check box on the **Status Configuration** form.
- Set the node group status using the thresholds set for each target status. For example, the default threshold for the target status of Minor is 20%. NNMi sets the status of the node group to Minor when 20% (or more) of the nodes in the node group have Minor status. To use this approach, clear the **Propagate Most Severe**

Status check box on the **Status Configuration** form. You can change the percentage thresholds for the target thresholds on the **Node Group Status Settings** tab of this form.

Because status calculations for large node groups can be resource-intensive, node group status calculation is off by default for new installations of NNMi. (Upgrades from NNMi 8.x retain the prior status calculation settings.) You can enable status calculation with the **Calculate Status** check box on the **Node Group** form for each node group.

Interface Groups

Interface groups filter interfaces within nodes by IFTYPE or by other attributes, such as ifAlias, ifDescr, ifName, ifIndex, IP address, and so forth. Interface groups carry no hierarchy or containment, although you can further qualify membership based on the node group for the node hosting the interface.

Interface groups can be filtered on custom capabilities and attributes similarly to node groups.

Qualifications for interface groups are AND'd together within and across tabs.

Node/Interface/Address Hierarchy

NNMi assigns monitoring settings in the following manner:

- 1 **Interface Settings**—NNMi monitors each of the node's interfaces and IP addresses based on the first matching **Interface Settings** definition. The first match is the **Interface Settings** definition with the lowest ordering number.
- 2 **Node Settings**—NNMi monitors each node and each previously unmatched interface or IP address based on the first matching **Node Settings** definition. The first match is the **Node Settings** definition with the lowest ordering number.



Child node groups are included in the ordering hierarchy. If the parent node group has a lower ordering number (for example, parent=10, child=20), then the monitoring configuration specified for the parent node group also applies to the nodes in the child node group. To override a parent node group monitoring configuration, set the ordering number for the child node group to a number that is lower than the parent (for example, parent=20, child=10).

- 3 **Default Settings**—If no match is found for a node, interface, or IP address in [step 1](#) or [step 2](#), NNMi applies the default monitoring configuration settings.

Stop Everything and Start Over Again

If you want to completely restart discovery and redo all of the NNMi configuration, or if the NNMi database has become corrupted, you can reset the NNMi configuration and database. This process deletes *all* of the NNMi configuration, topology, and incidents.

For information about the commands identified in this procedure, see the appropriate reference pages, or the UNIX manpages.

Follow these steps:

- 1 Stop the NNMi services:

```
ovstop -c
```

- 2 Optional. Because this procedure deletes the database, you might want to back up the existing database before proceeding:

```
nnmbackup.ovpl -type offline -target <backup_directory>
```

- 3 Optional. If you want to keep any of the current NNMi configuration, use the `nnmconfigexport.ovpl` command to output the NNMi configuration to an XML file.



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

- 4 Optional. Use the `nnmtrimincidents.ovpl` command to archive the NNMi incidents.
- 5 Drop and recreate the NNMi database.

- For the embedded database, run the following command:

```
nnmresetembdb.ovpl -nostart
```

- For an Oracle database, ask the Oracle database administrator to drop and recreate the NNMi database. Maintain the database instance name.

- 6 If you have installed iSPIs or stand-alone products that integrate with NNMi, reset those products to remove the old topology identifiers. For specific procedures, see the product documentation.
- 7 Start the NNMi services:

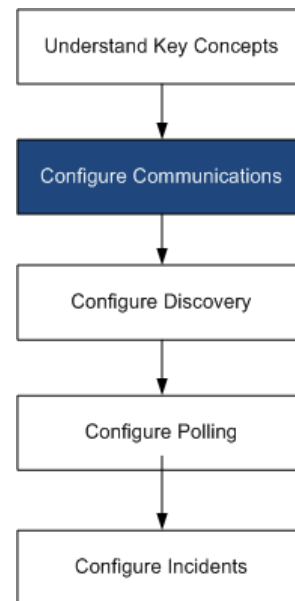
```
ovstart -c
```

NNMi now has only the default configurations as if you had just installed the product on a new system.

- 8 Start configuring NNMi. Do one of the following:

- Use the Quick Start Configuration Wizard.
- Enter information into the **Configuration** workspace in the NNMi console.
- Use the `nnmconfigimport.ovpl` command to import some or all of the NNMi configuration that you saved in [step 3](#).

NNMi Communications



HP Network Node Manager i Software uses both Simple Network Management Protocol (SNMP) and Internet Control Message Protocol (ICMP ping) to discover devices and to monitor device status and health. To establish viable communication in your environment, you configure NNMi with the access credentials and appropriate timeout and retry values for different devices and areas of your network. You can disable a protocol in some areas of your network to reduce traffic or to respect firewalls.

The communication values that you configure form the foundation of NNMi discovery and state polling. NNMi applies the appropriate values for each device when making queries for discovery or polling. Thus, if you configure NNMi to disallow SNMP communication within some region of your network, neither NNMi discovery nor NNMi state polling can send SNMP requests to that region.

This chapter contains the following topics:

- [Concepts for Communications](#)
- [Plan Communications](#)
- [Configure Communications](#)
- [Evaluate Communications](#)
- [Tune Communications](#)

Concepts for Communications

NNMi uses SNMP and ICMP primarily in a request-response manner. Responses to ICMP ping requests verify address responsiveness. Responses to SNMP requests for specific MIB objects provide more comprehensive information about a node.

The following concepts apply to NNMi communications configuration:

- [Levels of Communication Configuration](#)
- [Network Latency and Timeouts](#)
- [SNMP Access Control](#)
- [SNMP Version Preferences](#)
- [Management Address Preferences](#)
- [Polling Protocols](#)

Levels of Communication Configuration

NNMi communication configuration provides the following levels:

- Specific nodes
- Regions
- Global defaults

At each level you can configure access credentials, timeout and retry values, ICMP and SNMP protocol enablement, and SNMP access settings. If you leave settings blank at one level, NNMi applies the next level of defaults.

When communicating with a given node, NNMi applies the configuration settings as follows:

- 1 If the node matches a **specific node** configuration, NNMi uses any communication values in that configuration.
- 2 If any settings are not yet defined, NNMi determines whether the node belongs to any **regions**. Because regions might overlap, NNMi uses the matching region with the lowest ordering number. NNMi uses the values specified for that region to fill in the blanks left from the applicable specific node setting (if any). The settings for additional regions are not considered.
- 3 If any settings are still not yet defined, NNMi uses the **global default** settings to fill in the remaining blanks.

The values used for ICMP and SNMP communication with a particular device might be built up cumulatively until all required settings are determined.

Network Latency and Timeouts

Normal network latency influences the amount of time the NNMi management server must wait to get answers to ICMP and SNMP queries. Different areas of a network customarily have different turnaround times. For example, the local network where the NNMi management server resides could provide nearly instantaneous response, while responses from a device in a remote geographical region accessed through a dial-up wide area link would typically take much longer. In addition, heavily-loaded

devices might be too busy to respond to ICMP or SNMP queries immediately. When deciding which timeout and retry settings to configure, consider these latency concerns.

You can configure specific timeout and retry settings for both network regions and specific devices. The settings you choose determine how long NNMi waits for an answer and how many times NNMi requests data before abandoning the request when no answer is received.

For each request retry, NNMi adds the configured timeout value to the previous timeout value. Thus, the pause gets longer between each retry. For example, when NNMi is configured to use timeout of 5 seconds and three retries, NNMi waits 5 seconds for a response to the first request, 10 seconds for a response to the second request, and 15 seconds for a response to the third request before giving up until the next polling cycle.

SNMP Access Control

Communication with SNMP agents on managed devices requires access control credentials:

- SNMPv1 and SNMPv2c

A community string in each NNMi request must match a community string configured in the responding SNMP agent. All communication passes through the network in clear text (no encryption).

- SNMPv3

Communication with the SNMP agent complies with the user-based security model (USM). Each SNMP agent has a list of configured user names and their associated authentication requirements (the authentication profile). Formatting of all communication is controlled through configuration settings. NNMi SNMP requests must specify a valid user and follow the authentication and privacy controls configured for that user.

- Authentication protocol uses hash-based message authentication code (HMAC) using your choice of either the message-digest algorithm 5 (MD5) or the secure hash algorithm (SHA).
- Privacy protocol uses no encryption or the data encryption standard - cipher block chaining (DES-CBC) symmetric encryption protocol.

NNMi supports the specification of multiple SNMP access control credentials for a region of your network (defined through IP address filters or hostname filters). NNMi attempts communication with a device in that region by trying all configured values at a given SNMP security level in parallel. You can specify the minimum SNMP security level that NNMi uses in that region. NNMi uses the first value returned by each node (response from the device's SNMP agent) for discovery and monitoring purposes.

SNMP Version Preferences

The SNMP protocol itself has evolved over the years from version 1 to version 2(c) and now version 3, with increasing security capabilities (among others). NNMi can handle any or a mix of all versions in your network environment.

The first SNMP response NNMi receives for a particular node determines the communication credentials and SNMP version used by NNMi for communication with that node.



The SNMP version selection for a node plays a role in NNMi accepting traps from that node:

- If the source node or source object of the incoming trap has been discovered by NNMi using SNMPv3, NNMi accepts incoming SNMPv1, SNMPv2c, and SNMPv3 traps.
- If the source node or source object of the incoming trap has been discovered by NNMi using SNMPv1 or SNMPv2c, NNMi discards incoming SNMPv3 traps.

You specify the minimum level of SNMP version and security settings that are acceptable in each area of your network. The options for the SNMP Minimum Security Level field are as follows:

- **Community Only (SNMPv1 only)**—NNMi attempts to communicate using SNMPv1 with the configured values for community strings, timeouts, and retries. NNMi does not try any SNMPv2c or any SNMPv3 settings.
- **Community Only (SNMPv1 or v2c)**—NNMi attempts to communicate using SNMPv2c with the configured values for community strings, timeouts, and retries. If there is no response to any community string using SNMPv2c, NNMi attempts to communicate using SNMPv1 with the configured values for community strings, timeouts, and retries. NNMi does not try any SNMPv3 settings.
- **Community**—NNMi attempts to communicate using SNMPv2c with the configured values for community strings, timeouts, and retries. If there is no response to any community string using SNMPv2c, NNMi attempts to communicate using SNMPv1 with the configured values for community strings, timeouts, and retries. If none work, NNMi tries SNMPv3.
- **No Authentication, No Privacy**—For users with no authentication and no privacy, NNMi attempts to communicate using SNMPv3 with the configured values for timeouts and retries. If none work, NNMi tries users with authentication and no privacy followed by users with authentication and privacy, if necessary.
- **Authentication, No Privacy**—For users with authentication and no privacy, NNMi attempts to communicate using SNMPv3 with the configured values for timeouts and retries. If none work, NNMi tries users with authentication and privacy.
- **Authentication, Privacy**—For users with authentication and privacy, NNMi attempts to communicate using SNMPv3 with the configured values for timeouts and retries.

Management Address Preferences

A node's **management address** is the address NNMI uses to communicate with the node's SNMP agent. You can specify the management address for a node (in the specific node settings), or you can let NNMI choose an address from the IP addresses associated with the node. You can fine-tune this behavior in the discovery configuration settings by excluding certain addresses from discovery. For information about how NNMI determines the management address, see *Node Form* in the NNMI help.

NNMI discovers and monitors devices on an ongoing basis. *After the first NNMI discovery cycle*, the **Enable SNMP Address Rediscovery** field controls NNMI behavior when previously discovered SNMP agents quit responding (for example, when you reconfigure the device's SNMP agent).

- If the **Enable SNMP Address Rediscovery** check box is selected, NNMI retries any configured values in search of one that works.
- If the **Enable SNMP Address Rediscovery** check box is cleared, NNMI reports the device as “Down” and does not attempt to find another communication configuration setting for that device.



The **Enable SNMP Address Rediscovery** check box is available at all levels of communication configuration.



The **Discover Any SNMP Device** and **Non-SNMP Devices** auto-discovery rule configuration fields influence the way NNMI uses SNMP. For more information, see *Configure Basic Settings for the Auto-Discovery Rule* in the NNMI help.

Polling Protocols

You can prevent NNMI from using SNMP or ICMP in portions of your network (for example, when firewalls in your infrastructure prohibit ICMP or SNMP traffic).

Disabling ICMP traffic to the devices in an area of the network has the following results in NNMI:

- The optional auto-discovery rule ping sweep feature cannot locate additional nodes in that region of your network. All nodes must either be seeded or available through answers to various MIB object requests, such as neighbor's ARP cache, Cisco Discovery Protocol (CDP), or Extreme Discovery Protocol (EDP). Wide area network devices might be missed unless you seed every one of them.
- The State Poller cannot monitor devices that are not configured to respond to SNMP requests. (However, if the device responds to SNMP, State Poller does not use ICMP.)
- Operators cannot use **Actions > Ping** to check device reachability during troubleshooting.

Disabling SNMP traffic to the devices in an area of the network has the following results in NNMI:

- Discovery cannot gather any information about the devices except that they exist. All devices receive the `No SNMP` device profile.
- Discovery cannot find additional neighboring devices through queries. All devices must be directly seeded.

- Discovery cannot gather connectivity information from the devices, so they appear unconnected on NNMi maps.
- For devices with the `No SNMP` device profile, the State Poller respects the defaults of monitoring that device using only ICMP (ping).
- The State Poller cannot gather component health or performance data from the devices.
- The Causal Engine cannot contact the devices to perform neighbor analysis and locate the root cause of incidents.

Plan Communications

Make decisions in the following areas:

- [Default Communication Settings](#)
- [Communication Configuration Regions](#)
- [Specific Node Configurations](#)
- [Retry and Timeout Values](#)
- [Active Protocols](#)
- [Multiple Community Strings or Authentication Profiles](#)

Default Communication Settings

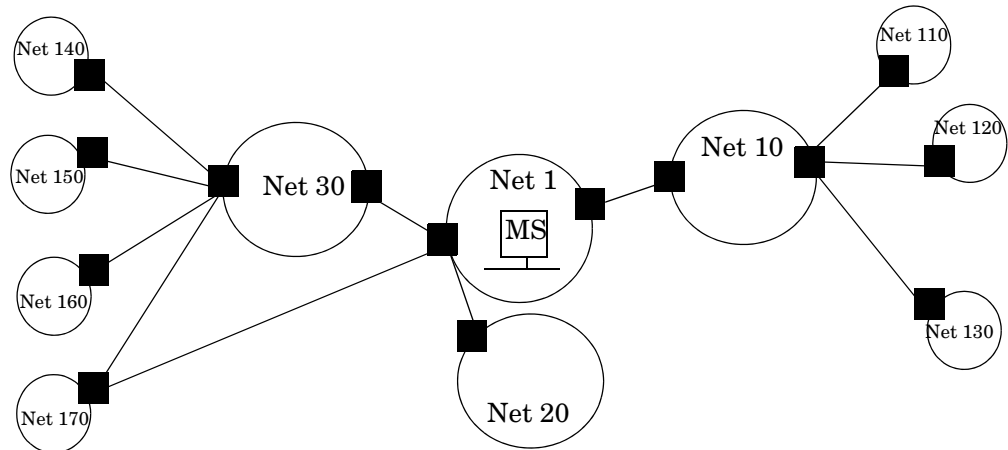
Because NNMi uses default values to complete any configuration settings that were not specified for the applicable region or specific node, set defaults to be reasonable for the majority of your network.

- Are there commonly-used community strings that NNMi should try?
- What default timeout and retry values are reasonable in your network?

Communication Configuration Regions

Regions represent areas of the network where similar communication settings make sense. For example, the local network around the NNMi management server usually returns responses very quickly. Areas of your network that are multiple hops away typically take longer to respond.

You do not need to configure each subnet or area of your network. You can combine areas into one region based on similar lag times. Consider the following network map:



For timeout and retry purposes, you might want to configure the following regions:

- Region A for Net 1
- Region B to include Net 10, Net 20, and Net 30
- Region C for the more distant outlying networks

You would decide how best to group Net 170, depending on whether traffic management configuration is set to prefer the one-hop or two-hop path from the NNMi management server.

Regions are also used to group devices with similar access credentials. If all routers in your network use the same community string (or a small set of possible community strings) and you can identify the routers with a naming convention (for example, `rtrnnn.yourdomain.com`), you can configure a region containing all routers so that they are handled similarly. If you cannot use a wildcard to group the devices, you can configure each as a specific node.

Plan your region configurations so that you can apply the same timeout and retry value and access credential configurations to all nodes in a region.

Region definitions can overlap, and a device might qualify for multiple regions. NNMi applies the settings from the region with the lowest ordering number (and no other matching regions).

Specific Node Configurations

For any device with unique communication configuration requirements, use the specific node settings to specify the communication settings for that node. Example uses of specific node settings include the following:

- A node that might not respond well to SNMPv2c/SNMPv3 GetBulk requests
- A node whose name does not match the name pattern of other similar nodes

Retry and Timeout Values

Configuring longer timeouts and more retries can result in more responses from devices that are busy or distant. This higher response rate eliminates false down messages. However, it also lengthens the time to determine that actual down devices require attention. Finding the balance for each area of your network is important and may require a period of testing and adjusting values in your environment.

To get an idea of current lag time for each hop, do the following:

- *Windows*: Run a `tracert` to a device in each network area.
- *UNIX*: Run a `traceroute` to a device in each network area.

Active Protocols

You have two opportunities to control the type of traffic NNMi generates when communicating with devices in your network: communication and monitoring configuration settings. Use the communication settings when firewalls in your infrastructure prohibit ICMP or SNMP traffic. Use monitoring settings to fine tune protocol usage when you do not need a particular subset of data about devices. If either communication or monitoring settings disable a protocol for a device, NNMi does not generate that type of traffic to the device.



Disabling SNMP communication significantly compromises the NNMi status and health monitoring of your network.

Note whether each region or specific device should receive ICMP traffic.

You do not need to explicitly disable SNMP communication with devices for which you do not supply access credentials. By default, NNMi assigns those devices to the `No SNMP` device profile and monitors them using ICMP only.

Multiple Community Strings or Authentication Profiles

Plan the community strings and authentication profiles to be tried for each area of your network. For the default and region settings, you can configure multiple community strings and authentication profiles to be tried in parallel.



While trying probable community strings, NNMi queries might cause devices to generate authentication failures. Inform your operations department that authentication failures may safely be ignored while NNMi completes its initial discovery. Alternatively, you can minimize the number of authentication failures by configuring your regions (and the associated community strings and authentication protocols to try) as tightly as possible.

If your environment uses SNMPv1 or v2c *and* SNMPv3, determine the minimum acceptable security level for each region.

For regions where SNMPv1 or v2c access is acceptable, gather the community strings in use within the region and any unique community strings required by specific devices.

For regions containing SNMPv3-accessible devices, determine the minimum acceptable default authentication profiles, the authentication profiles appropriate for each region, and the unique authentication credentials in use on specific devices (if any).

Configure Communications

After reading the information in this section, see *Configuring Communication Protocol* in the NNMi help for specific procedures.

- ▶ It is a good idea to save a copy of the existing configuration before you make any major configuration changes. For more information, see [Best Practice: Save the Existing Configuration](#) on page 38.

Configure the following areas of communication:

- Default settings
- Region definitions and their settings
- Specific node settings

For specific nodes, you can enter node settings through the NNMi console or through a configuration file.

- ▶ **Save and Close** all **Communication Configuration** forms all of the way back to the NNMi console to implement your changes.

Best practice

Double-check the ordering numbers for the defined regions. If a node qualifies for membership in multiple regions, NNMi applies the settings from the region with the lowest ordering number to that node.

Evaluate Communications

This section lists ways to evaluate the progress and success of the communications settings. Most of these tasks can be completed only after discovery has completed.

Consider the following:

- [Are All Nodes Configured for SNMP?](#)
- [Is SNMP Access Currently Available for a Device?](#)
- [Is the Management IP Address Correct?](#)
- [Is NNMi Using the Correct Communications Settings?](#)
- [Do the State Poller Settings Agree with the Communication Settings?](#)

Are All Nodes Configured for SNMP?

- 1 Open the **Nodes** inventory view.
- 2 Filter the **Device Profile** column to contain the string `No SNMP`.
 - For each of the devices that you want to manage, configure communication settings for the specific node. Alternatively, you can expand a region to include the node and update the access credentials.
 - If the communication settings are correct, verify that the SNMP agent on the device is running and properly configured (including ACLs).

Is SNMP Access Currently Available for a Device?

- 1 Select the node in an inventory view.
- 2 Select **Actions > Status Poll** or **Actions > Configuration Poll**.

If the results show any SNMP values, communication is operational.

You can also test communication from the command line with the `nnmsnmpwalk.ovpl` command. For more information, see the `nnmsnmpwalk.ovpl` reference page, or the UNIX manpage.

Is the Management IP Address Correct?

To determine which management address NNMi has selected for a device, follow these steps:

- 1 Select the node in an inventory view.
- 2 Select **Actions > Communication Settings**.
- 3 In the **Communication Configuration** window, verify that the management address of the SNMP agent listed in the Active SNMP Agent Settings list is correct.

Is NNMi Using the Correct Communications Settings?

Missing or incorrect SNMP community strings can result in incomplete discovery or can negatively impact the discovery performance.

To verify the communication settings configured for a device, use the `nnmcommconf.ovpl` command or follow these steps:

- 1 Select the node in an inventory view.
- 2 Select **Actions > Communication Settings**.
- 3 In the **Communication Configuration** window, verify that the values listed in the SNMP configuration settings table are the settings you want NNMi to use for this node.

If the communication settings are not correct, use the source information in the SNMP configuration settings table as a starting point for fixing the problem. You might need to change the configuration or the ordering number of a region or specific node.

Do the State Poller Settings Agree with the Communication Settings?

Even if the communication settings allow protocol traffic to an area of your network, that type of traffic might be disabled in the monitoring settings. To determine whether the settings are being overridden:

- 1 Select the node in an inventory view.
- 2 Select **Actions > Monitoring Settings**.

If either the Monitoring Settings or the Communication Settings disable a type of traffic to the device, that traffic will not be sent from NNMi.

Tune Communications

Reduce authentication failures

If NNMi is generating too many authentication traps during discovery, configure smaller regions or specific nodes with smaller groups of access credentials for NNMi to try.

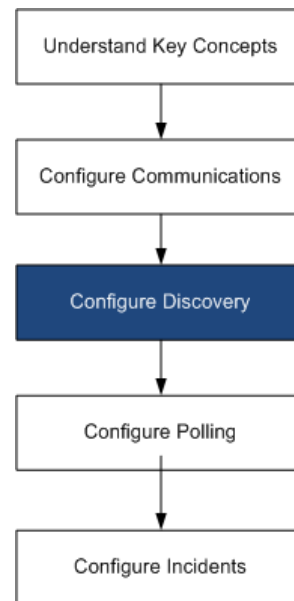
Tune timeouts and retries

When NNMi attempts to contact a device using SNMP during discovery, the communication configuration determines whether NNMi can gather the necessary device information. When the communication configuration does not include the correct SNMP community strings, or if NNMi is discovering non-SNMP devices, NNMi uses the configured settings for SNMP timeouts and retries. In this case, large timeout values or a high number of retries can negatively impact the overall performance of discovery. If your network contains devices that you know respond slowly to SNMP/ICMP requests, consider using the **Regions** or **Specific Node Settings** tabs on the **Communication Configuration** form to fine tune the timeout and retry values for just these devices.

Reduce default community strings

Having a large number of default community strings can negatively impact discovery performance. Instead of entering many default community strings, fine tune the community string configuration for particular areas of your network by using the **Regions** or **Specific Node Settings** tabs on the **Communication Configuration** form.

NNMi Discovery



One of the most important network management tasks is keeping your view of the network topology current. HP Network Node Manager i Software discovery populates the topology inventory with information about the nodes in your network. NNMi maintains this topology information through ongoing spiral discovery, which ensures that root cause analysis and the troubleshooting tools provide accurate information regarding incidents.

This chapter provides information to help you configure NNMi discovery. For an introduction to how discovery works and for detailed information about how to configure discovery, see *Discovering Your Network* in the NNMi help.

If you have experience working with NNM 6.x/7.x and you want to understand how discovery has changed in NNMi 9.00, see [Network Discovery](#) on page 313 for a high-level overview of the differences.

This chapter contains the following topics:

- [Concepts for Discovery](#)
- [Plan Discovery](#)
- [Configure Discovery](#)
- [Evaluate Discovery](#)
- [Tune Discovery](#)

Concepts for Discovery

The NNMi default behavior of discovering only routers and switches allows you to focus your network management on the critical or most important devices. In other words, target the backbone of the network first. Generally, you should avoid managing end nodes (for example, personal computers or printers) unless the end node is identified as a critical resource. For example, database and application servers might be considered critical resources.

NNMi provides several ways to control what devices to discover and include in the NNMi topology. Your discovery configuration can be very simple, quite complex, or anywhere in between, depending on how your network is organized and what you want to manage with NNMi.



NNMi does not perform any default discovery. You must configure discovery before any devices appear in the NNMi topology.

Each discovered node (physical or virtually hosted) counts toward the license limit, regardless of whether NNMi is actively managing that node. The capacity of your NNMi license might influence your approach to discovery.

Status monitoring considerations might also influence your choices. By default, the State Poller only monitors interfaces connected to devices NNMi has discovered. You can override this default for some areas of your network, and you can discover the devices beyond the edge of your responsibility. (For information about the State Poller, see [NNMi State Polling](#) on page 71.)

NNMi provides two primary discovery configuration models:

- **List-based discovery**—Explicitly tell NNMi exactly which devices should be added to the database and monitored through a list of seeds.
- **Rule-based discovery**—Tell NNMi which areas of your network and device types should be added to the database, give NNMi a starting address in each area, and then let NNMi discover the defined devices.

You can use any combination of list-based and rule-based discovery to configure what NNMi should discover. Initial discovery adds these devices to the NNMi topology, and then spiral discovery routinely rediscovers the network to ensure that the topology remains current.

NNMi Derives Attributes through Device Profiles

As NNMi discovers devices, it uses SNMP to gather some attributes directly. One of the key attributes is the MIB II system object ID (`sysObjectID`). From the system object ID, NNMi derives additional attributes, such as vendor, device category, and device family.

During discovery, NNMi collects the MIB II system capabilities and stores them in the topology portion of the database. System capabilities are visible on the **Node** form. However, these capabilities are not used by any other portion of NNMi (specifically, monitoring configuration). NNMi uses the device category (from the device profile for the system object ID) to match devices into node groups. In node view tables, the **Device Category** column identifies the device category for each node.

NNMi ships with over 3600 device profiles for system object IDs that were available at the time of release. You can configure custom device profiles for the unique devices in your environment to map these devices to category, vendor, and so forth.

Plan Discovery

Make decisions in the following areas:

- [Select Your Primary Discovery Approach](#)
- [Auto-Discovery Rules](#)
- [Node Name Resolution](#)
- [Subnet Connection Rules](#)
- [Discovery Seeds](#)
- [Rediscovery Interval](#)
- [Do Not Discover Objects](#)

Select Your Primary Discovery Approach

Decide whether to do entirely list-based discovery, entirely rule-based discovery, or a combination of both approaches.

List-Based Discovery

With list-based discovery, you explicitly specify (as a discovery seed) each node that NNMi should discover.

Benefits of using only list-based discovery include:

- Provides very tight control over what NNMi manages.
- Simplest configuration.
- Good for fairly static networks.
- A good way to start using NNMi. You can add auto-discovery rules over time.

Disadvantages of using only list-based discovery include:

- NNMi does not discover new nodes as they are added to the network.
- You must provide the complete list of nodes to be discovered.

Rule-Based Discovery

With rule-based discovery, you create one or more auto-discovery rules to define the areas of the network that NNMi should discover and include in the NNMi topology. For each rule, you must provide one or more discovery seeds (by explicitly naming seeds or by enabling ping sweep), and then NNMi discovers the network automatically.

Benefits of using rule-based discovery include:

- Good for large networks. NNMi can discover a large number of devices based on minimal configuration input.
- Good for networks that change frequently. New devices that are added to the network are discovered without administrator intervention (assuming that each device is covered by an auto-discovery rule).

- Ensures that any new device added to your network is discovered to comply with service level agreements for managing new devices in a timely manner or security guidelines to flag unauthorized new devices.

Disadvantages of using rule-based discovery include:

- It is easier to run into license limitations.
- Depending on the structure of your network, tuning auto-discovery rules can be complex.
- If auto-discovery rules are very broad and NNMi discovers many more devices than you want to manage, you might want to delete the unneeded devices from NNMi topology. Node deletion can be time consuming.

Rule-based discovery
only

Auto-Discovery Rules

Auto-Discovery Rule Ordering

The value of an auto-discovery rule's **Ordering** attribute affects discovery ranges in the following ways:

- IP address ranges
 - If a device falls within two auto-discovery rules, the settings in the auto-discovery rule with the lowest ordering number applies. For example, if an auto-discovery rule excludes a set of IP addresses, then no other auto-discovery rules with higher ordering numbers process those nodes and the nodes within that range of addresses are not discovered unless they are listed as discovery seeds.
- System object ID ranges
 - If no IP address range is included in an auto-discovery rule, then the system object ID settings apply to all auto-discovery rules with higher ordering numbers.
 - If an IP address range is included in an auto-discovery rule, the system object ID range applies only within the auto-discovery rule.

Exclude Devices from Discovery

- To prevent discovery of certain object types, create an auto-discovery rule with a low ordering number that ignores the system object IDs that you do not want discovered. Do not include an IP address range in this rule. By giving this auto-discovery rule a low ordering number, the discovery process quickly passes by the objects that match this rule.
- The **Ignored by Rule** setting for an IP address range or a system object ID range affects that auto-discovery rule only. The devices included in an ignored range are available to be included in another auto-discovery rule.
- The addresses listed on the **Excluded IP Addresses** tab of the **Discovery Configuration** form apply to all auto-discovery rules. Unless they are configured as discovery seeds, these addresses are never added to the NNMi topology. (Discovery seeds are always discovered.)

Ping Sweep

You can use ping sweep to locate devices within the IP address ranges of the configured auto-discovery rules. For initial discovery, you might want to enable ping sweep for all rules. Doing so provides enough information to NNMi discovery that you do not need to configure discovery seeds.



Ping sweep works for subnets of 16 bits or smaller, for example, 10.10.*.*.

Ping sweeps are especially useful for discovering devices across a WAN that you do not control, such as an ISP network.



Firewalls often view ping sweeps as attacks on the network, in which case, a firewall might block all traffic from a device that emits ping sweeps.

Best practice

Enable ping sweep for small discovery ranges only.

Discovery Seeds for Auto-Discovery Rules

Provide at least one discovery seed per auto-discovery rule. The options for providing the seeds are as follows:

- Enter seeds on the **Discovery Seeds** tab of the **Discovery Configuration** form.
- Use the `nnmloadseeds.ovpl` command to load information from a seed file.
- Enable ping sweep for the rule, at least for initial discovery.

Best Practices for Auto-Discovery Rules

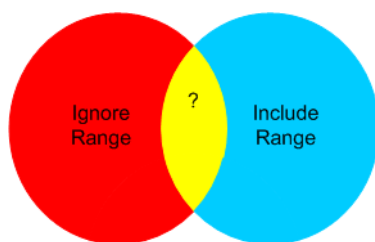
- Because NNMi automatically manages all discovered devices, use IP address ranges that closely match the areas of the network that you want to manage.
 - You can use multiple IP address ranges within an auto-discovery rule to restrict discovery.
 - You can add a large IP address range to an auto-discovery rule and then exclude some IP addresses from discovery within that rule.
- The system object ID range specification is a prefix, not an absolute value. For example, the range 1.3.6.1.4.1.11 is the same as 1.3.6.1.4.1.11.*.

Examples

Discovery Rule Overlap

Figure 1 shows two discovery ranges that overlap. The circle on the left represents an IP address range or a system object ID range to be ignored by NNMi discovery. The circle on the right represents an IP address range or a system object ID range to be discovered and included in the NNMi topology. The overlapping region might be included or ignored by discovery, depending on the ordering of these auto-discovery rules.

Figure 1 Overlapping Discovery Ranges



Limit Device Type Discovery

To discover all HP devices in your network that are not printers, create one auto-discovery rule with a range to include the HP enterprise system object ID (1.3.6.1.4.1.11). In this auto-discovery rule, create a second range to ignore the system object IDs of HP printers (1.3.6.1.4.1.11.2.3.9). Leave the IP address range unset.

Node Name Resolution

By default, NNMi attempts to identify a node in the following order:

- 1 Short DNS name
- 2 Short sysName
- 3 IP Address

The following scenarios describe situations in which you might want to change the default order for node name resolution:

- If your organization is dependent on others to update the DNS configuration, you might set a policy of defining the sysName for each new device as it is added to the network. In this case, set select sysName as the first choice for node name resolution so that NNMi can discover the new device as soon as it is deployed in the network. (Maintain the sysName over the life of the device.)
- If your organization does not set or maintain the sysName for managed devices, select sysName as the third option for node name resolution.

Best practice

If you use the full or short DNS name as the primary naming convention, confirm that you have forward and reverse DNS resolution from the NNMi management server to all managed devices.



When the full DNS name is the naming convention, labels on the topology maps can be long.

Best practice

NNMi selects the lowest loopback address as the management address for Cisco devices, so put DNS resolution on the lowest loopback address for each Cisco device. (NNMi 8.0x selects the highest loopback address as the management address.)

Subnet Connection Rules

List-based discovery only

For list-based discovery, NNMi uses the subnet connection rules to detect connections that span a WAN. NNMi evaluates the subnet membership of the device it has discovered on each end of a probable connection (by examining their IP addresses and subnet prefixes) and looks at subnet connection rules for a match.

- Rule-based discovery only
- When auto-discovery rules are enabled and NNMi finds a device configured with a subnet prefix between /28 and /31:
- 1 NNMi checks for an applicable subnet connection rule.
 - 2 If a match is found, NNMi uses each valid address in the subnet as a hint and attempts a discovery on that address.

Best practice Use the default connection rules. Only modify them if you have a problem.

Discovery Seeds

List the devices to use as discovery seeds.

Best practice One of the NNMi rules for selecting the preferred management IP address specifies using the first discovered IP address as the management address. You can influence NNMi by configuring the preferred IP address as the seed address.

Best practice For Cisco devices, use a loopback address as the discovery seed because loopback addresses are more reliably reachable than other addresses on a device. Ensure that DNS is correctly configured to resolve the device hostname to the loopback address.

- List-based discovery only
- For list-based discovery, list all devices that you want NNMi to manage. You might be able to export this list from asset management software or from some other tool.
- Because NNMi does not automatically add any devices to this list, ensure that the list includes every device for which you have responsibility or which influences your monitoring and status calculations.

- Rule-based discovery only
- Discovery seeds are optional for rule-based discovery:
- If ping sweep is enabled for an auto-discovery rule, you do not need to specify a seed for that rule.
 - For each auto-discovery rule with ping sweep disabled, identify at least one seed per rule. If a rule includes multiple IP address regions, you might need a seed in each routable region because routers do not keep ARP entries across WAN links.

Best practice For the most complete rule-based discovery, use routers, not switches, as discovery seeds because routers generally have much larger ARP caches than do switches. A core router connected to a network that you want to discover is an excellent choice for a discovery seed.

Rediscovery Interval

NNMi rechecks the configuration information from each device in the database according to the configured rediscovery interval. In addition, NNMi collects the ARP cache from each router covered by an auto-discovery rule and looks for new nodes on the network.

Any change in the communication-related configuration of a device, such as interface renumbering, automatically triggers NNMi to update its data for that device and its neighbors.

The following changes do not trigger an automatic rediscovery; devices are updated only at the configured rediscovery interval:

- Changes within a node (for example, firmware upgrade or system contact).
- New nodes added to the network.

Select the rediscovery interval to match the level of change in the network. For a highly-dynamic network, you might want to use the minimum interval of 24 hours. For more stable networks, you can safely extend that period.

Do Not Discover Objects

In NNMi, there are three ways that you can configure NNMi to disregard certain objects:

- On the **Communication Configuration** form, you can turn off ICMP and/or SNMP communication at various levels: globally, for communication regions, or for specific hostnames or IP addresses. For information about the impacts of disabling one or both of these protocols, see [Polling Protocols](#) on page 49.
- On the **Discovery Configuration** form, you can set up an auto-discovery rule that instructs NNMi to never gather hints from certain IP addresses or SNMP system object IDs. Nodes matching the criteria still appear on the map and in the database, but spiral discovery does not extend to the neighboring devices beyond those IP addresses or object types.
- On the **Discovery Configuration** form, you can set up an auto-discovery rule that instructs NNMi to exclude certain IP address ranges and/or specific IP addresses from the database. Spiral discovery does not display those addresses on any node's list of addresses or use those addresses when establishing connections between devices, so NNMi never monitors the health of those addresses.

Configure Discovery

This section lists configuration tips and provides some configuration examples. After reading the information in this section, see *Configure Discovery* in the NNMi help for specific procedures.

- ▶ Because NNMi launches discovery from seeds as soon as you **Save and Close** the **Discovery Seed** form, ensure that you do the following before you configure seeds:
 - Complete all communication configuration.
 - Complete all auto-discovery rules (if any).
 - Configure subnet connection rules.
 - Configure name resolution preferences.
 - **Save and Close** all the way back to the console.
- ▶ It is a good idea to save a copy of the existing configuration before you make any major configuration changes. For more information, see [Best Practice: Save the Existing Configuration](#) on page 38.

Tips for Configuring Auto-Discovery Rules

- As you define a new auto-discovery rule, check each setting carefully. For a new rule, auto-discovery is enabled by default, IP address ranges are included by default, and system object ID ranges are *ignored* by default.

Tips for Configuring Seeds

- If you already have a file that lists the nodes to be discovered, format this information as a seed file and use the `nnmloadseeds.ovpl` command to import the node list into NNMi.
- In the seed file, specify IP addresses as a way of influencing the IP address that NNMi chooses as the management address. (If you use hostnames, DNS provides the IP address for each node.)
- A good format for each entry in the seed file is shown here:

```
IP_address # node name
```

This format is easy for both NNMi and human readers.

- For maintenance purposes, it is better to use only one seed file. Add nodes as needed and then rerun the `nnmloadseeds.ovpl` command. NNMi discovers the new nodes but does not re-evaluate the existing nodes.
- Removing a node from the seed file does not remove it from the NNMi topology. Delete the node directly in the NNMi console.
- Deleting a node from a map or inventory view does not delete the seed.
- If you want NNMi to rediscover a node, delete that node from a map or inventory view *and* from the **Discovery Seeds** tab on the **Discovery Configuration** form in the NNMi console, and the re-enter the node in the NNMi console, or run the `nnmloadseeds.ovpl` command.
- Completely configure a discovery rule *before* you specify a seed for that rule. That is, click **Save and Close** on the **Discovery Configuration** form. (Although it appears to be connected, the **Discovery Seeds** tab is a separate form that is not part of the **Discovery Configuration** form in the database model. As a result, when you save the information about the **Discovery Seeds** tab, NNMi updates the seed configuration immediately.)

Rule-based discovery
only

Evaluate Discovery

This section lists ways to evaluate the progress and success of discovery.

Follow the Progress of Initial Discovery

NNMi discovery is dynamic and ongoing; it is never complete, so you will never see a “discovery completed” message. The process of initial discovery and connection takes some time. The following items suggest ways to gauge the progress of initial discovery:

- On the **Database** tab of the **System Information** window, watch for the node count to reach the expected level and stabilize. This window does not refresh automatically. During initial discovery, open the **System Information** window several times.
- On the **Discovery Configuration** form, look at the **Discovery Seeds** tab. Refresh this tab until all seeds show the `Node created` results, which indicates that the device has been added to the topology database. This result does *not* indicate that NNMi has gathered all information from the device and processed its connectivity.
- Open the **Node** form for representative nodes. When the **Discovery State** field transitions to `Discovery Completed`, NNMi has gathered the node’s basic characteristics as well as the node’s ARP cache and discovery protocol neighbors, if applicable. This state does *not* indicate that NNMi has completed connectivity analysis for the device.
- In the **Nodes** inventory view, scan to see that key devices are present from different areas of your network.
- Open the **Layer 2 Neighbor View** for representative nodes to determine whether connectivity analysis has completed for that area.
- Review the **Layer 2 Connections** and **VLANs** inventory views to gauge the progress of layer 2 processing.

Were All Seeds Discovered?

- 1 Open the **Discovery Configuration** form.
- 2 On the **Discovery Seeds** tab, sort the list of nodes by the **Discovery Seed Results** column. For any node in an error state, consider the following:
 - Failed discovery due to an unreachable node or unresolved DNS name—For these types of failures, verify network connectivity to the node and check for accurate DNS name resolution. To work around DNS issues, use the IP address to seed the node or include the hostname in a `hostnolookup.conf` file.
 - License node count exceeded—This scenario occurs when the number of devices already discovered reached your license limit. You can either delete some discovered nodes or purchase additional node pack licenses.
 - Node discovered but no SNMP response—SNMP communication problems can occur for seeded devices as well as devices that are discovered through auto-discovery. For more information, see [Evaluate Communications](#) on page 53.

Do All Nodes Have a Valid Device Profile?

- 1 Open the **Nodes** inventory view.
- 2 Filter the **Device Profile** column to contain the string `No Device Profile`.
- 3 If a node is discovered but has no device profile, add a new device profile (from **Configuration > Device Profiles**), and then perform a configuration poll on the node to update its data.

Were All Nodes Discovered Properly?

Examine the data in the **Nodes** inventory view. If any nodes do not have a management address, check the communication settings for those nodes as described in [Are All Nodes Configured for SNMP?](#) on page 53.

If any expected nodes are missing from the **Nodes** inventory view, check the following:

- On each missing node, verify that the discovery protocol (for example, CDP) is correctly configured.
- If a missing node is on a WAN, enable ping sweep for the auto-discovery rule that includes that node.

List-based discovery
only

Auto-Discovery Rules

If you see unexpected discovery results, re-evaluate the auto-discovery rules.

When NNMi discovery finds an address hint, it uses the first matching rule to determine if a node should be created. If no rules are matched, NNMi discovery discards the hint. The ordering number for auto-discovery rules determines the order in which the auto-discovery rule configuration settings are applied.

For each auto-discovery rule, check the following settings:

- **Discover Included Nodes** must be enabled for auto-discovery to occur for the rule.
- Verify that the following settings are correct for the type of nodes you want discovered for the rule:
 - **Discover Any SNMP Device**
 - **Discover Non-SNMP Devices**

Remember that only routers and switches are discovered by default and non-SNMP nodes are *not* discovered. Enabling these settings without considering your environment can result in NNMi discovering more nodes than intended.

IP Address Ranges

The IP address of a discovery hint must match an **Include in Rule** entry in the IP address range list. If there are no included IP address ranges in an auto-discovery rule, then all address hints are considered a match. (For this case, see [Tips for Configuring Auto-Discovery Rules](#) on page 64.) Additionally, the hint must *not* match any entry marked **Ignored by Rule**. If all checks successfully match, this rule's configuration is used for handling the hint.

- If you are not discovering some expected devices, check your configured IP ranges to ensure that the IP addresses for those devices are included in a range and not ignored by a rule with a lower ordering number.

- If you are discovering more devices that you want, modify the include ranges or add ignored ranges for the IP addresses of the devices that you do not want discovered. Also, determine if **Discover Any SNMP Device** is enabled.

System Object ID Ranges

The system object ID (OID) from a discovery hint must match an **Include in Rule** entry in the system object ID ranges list. If there are no included system object ID ranges in an auto-discovery rule, then all object IDs are considered a match. Additionally, the OID must not match any entry marked **Ignored by Rule**. If all checks successfully match, this rule's configuration is used for handling the hint.

- Use the system object ID ranges to either expand auto-discovery to include more than the default routers and switches, or to exclude specific routers and switches.
- Each node must match both the IP address range and the system object ID range specified before it is discovered and added to the topology database.

Are All Connections and VLANs Correct?

NNMi creates Layer 2 connections and VLANs as a separate step after devices are added to the topology. Give NNMi plenty of time for initial discovery before evaluating connections and VLANs.

To evaluate Layer 2 connectivity, create a node group for each network area of interest, and then display a topology map for that node group. (In the **Node Groups** inventory, select a node group, and then click **Actions > Node Group Map**.) Look for any nodes that are not connected to the other nodes in this map.

To evaluate VLANs, from the **VLANs** inventory view, open each **VLAN** form, and then examine the list of ports for that VLAN.

Rediscover a Device

- 1 Perform a configuration poll of the device.
- 2 Delete the device.

If the device is a seed, delete the seed, and then re-add the seed.

Tune Discovery

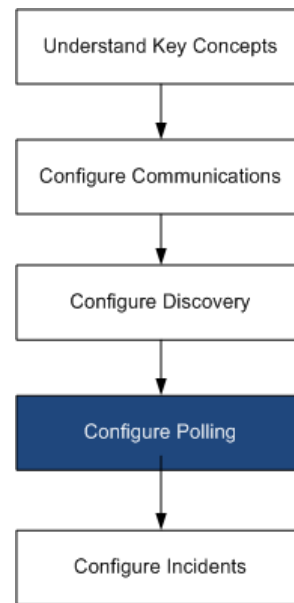
For general discovery performance, fine tune the discovery configuration to discover only critical and important devices.

- Filter by IP address range and/or system object ID.
- Limit discovery of non-SNMP devices and any SNMP devices (devices that are not switches or routers).

To delete one or more nodes from the NNMi database on the command line, use the `nmnode delete.ovpl` command. This command deletes nodes, but not seed definitions, from the NNMi database.

To delete one or more seed definitions from the NNMi database on the command line, use the `nmseeddelete.ovpl` command.

NNMi State Polling



This chapter provides information to help you expand and fine tune network monitoring by configuring the HP Network Node Manager i Software State Poller service. This chapter supplements the information in the NNMi help. For an introduction to how monitoring works and for detailed information about how to configure monitoring, see *Monitoring Network Health* in the NNMi help.

If you have experience working with NNM 6.x/7.x and you want to understand how monitoring has changed in NNMi 9.00, see [Status Monitoring](#) on page 316 for a high-level overview of the differences.

This chapter contains the following topics:

- [Concepts for State Polling](#)
- [Plan State Polling](#)
- [Configure State Polling](#)
- [Evaluate State Polling](#)
- [Tune State Polling](#)

Concepts for State Polling

This section provides a brief overview of network monitoring, including the order that the State Poller uses to evaluate polling groups. After reading the information in this section, continue to [Plan State Polling](#) on page 72 for more specific information.

As with network discovery, you should focus network monitoring on the critical or most important devices in the network. NNMi can only poll devices in the topology database. You control which network devices NNMi monitors, the type of polling to use, and the interval at which to poll.

You can use the interface and node settings on the **Monitoring Configuration** form to refine the status polling of devices, and to set different polling types and intervals for different classes, types of interfaces, and types of nodes.

You can configure State Poller data collection to be based on an ICMP (ping) response, or to be based on SNMP data. NNMi automatically handles the mapping from the type of data collection you enable to the actual MIB objects internally, significantly simplifying configuration.

As you plan polling configuration, you should carefully consider how to set up interface groups and node groups for the State Poller service. If you are new to the concept of *groups*, see [Node Groups](#) on page 39, and [Node/Interface/Address Hierarchy](#) on page 43 for overview information.

Order of evaluation

Because an interface or node may qualify for multiple groups, the State Poller applies the configured polling interval and polling type in a well-defined order of evaluation. For each object in the discovered topology:

- 1 If the object is an interface, State Poller looks for a qualifying interface group. Groups are evaluated from the lowest Order Number to the highest. The first matching group is used and evaluation stops.
- 2 If no interface group has captured the object, node groups are evaluated from lowest Order Number to highest. The first matching group is used and evaluation stops. Any contained interface which has not qualified for an interface group on its own characteristics inherits the polling settings from its hosting node.
- 3 For devices that are discovered but not included in any node or interface settings definitions, the global monitoring settings (on the **Default Settings** tab of the **Monitoring Configuration** form) establish the monitoring behavior.

Plan State Polling

This section provides information to plan for State Poller configuration, including a polling configuration checklist; and more detailed information to help you plan for monitoring, decide how to create polling groups, and determine what types of data should be captured during the polling process.

Polling Checklist

You can use the checklist below to plan for State Poller configuration.

- What can NNMi monitor?
- What are the logical groups for monitored items, based on object type, location, relative importance, or other criteria?
- How often should NNMi monitor each grouping?
- What data should be collected to capture information about the monitored item? This may include:
 - ICMP (ping) response
 - SNMP fault data
 - SNMP performance data if you have a license for one or more NNM iSPI Performance products
 - Additional SNMP Component Health data

Example polling configuration

To help you understand the polling configuration process, consider this example. Suppose that your network contains the latest proxy servers from ProximiT. You need to ensure that these devices can be reached, but you do not require SNMP monitoring of the proxy servers.

1 What can NNMi monitor?

Because you can only monitor what has been discovered, you configure auto-discovery rules to ensure that NNMi's database contains your ProximiT proxy servers. For more information on configuring discovery, see [NNMi Discovery](#) on page 57.

2 What are the logical groups for monitored items?

Obviously you'd like to group the ProximiT proxy servers together and apply the same monitoring settings to all of them. Because you are not doing interface (SNMP) monitoring for the devices, you do not need any interface groups.

You can also use this node group to filter views, to check the status of the proxy servers as a group, and to put the group OUT OF SERVICE to update firmware.

3 How often should NNMi monitor each group?

For your service level agreements, a five minute polling interval for the proxy servers is sufficient.

4 What data should be collected?

Here's where the monitoring configuration differs from other groups. For our ProximiT proxy server example, you enable ICMP fault monitoring and disable SNMP fault and polling monitoring. Without SNMP fault monitoring for the group, Component Health monitoring will not apply.

For more detailed planning information concerning these configuration choices, see the following topics:

- [What Can NNMi Monitor?](#) on page 73
- [Planning Groups](#) on page 75
- [Planning Polling Intervals](#) on page 77
- [Deciding What Data to Collect](#) on page 78

What Can NNMi Monitor?

By default, the NNMi State Poller uses SNMP polls to monitor the following:

- Interfaces that are connected to another known interface on an NNMi-discovered device.
- Router interfaces which host IP addresses.



In most cases, polling only connected interfaces provides sufficiently accurate root-cause analysis. Extending the set of monitored interfaces can impact polling performance.

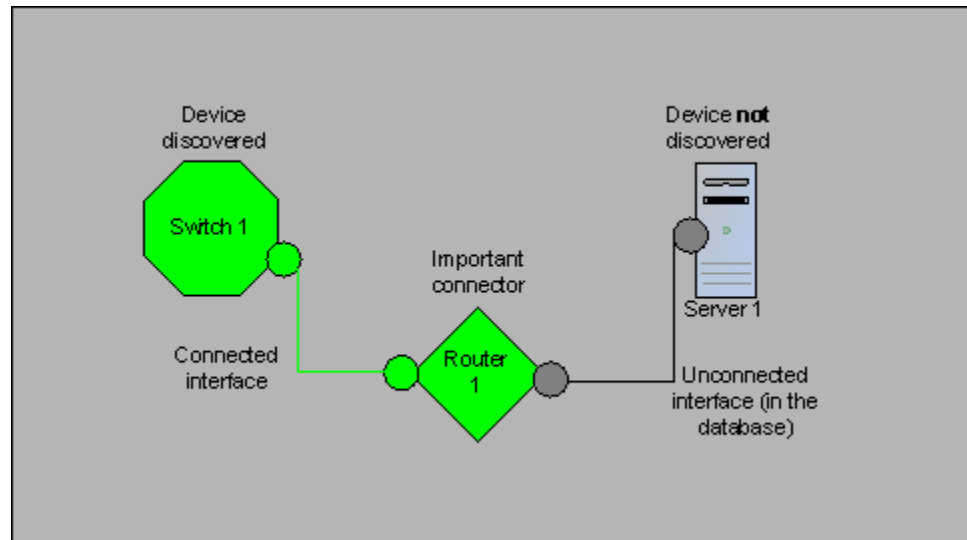
Extend monitoring

You can extend the monitoring to include the following:

- Unconnected interfaces. By default, the only unconnected interfaces that NNMI monitors are those that have IP addresses *and* are included in the **Routers** node group.



NNMI defines an unconnected interface as an interface that is not connected to another device discovered by NNMI, as shown below.



- Interfaces, such as router interfaces, that have an IP address.
- ICMP polling for devices that do not support SNMP. By default, ICMP polling is enabled for the **Non-SNMP Devices** node group.

Interfaces to Unmonitored Nodes

Sometimes, you need to know the status of an interface that connects to a device you do not manage directly. For example, you want to know whether the connection to an application or Internet server is up, but you might not be responsible for maintaining that server. If you do not include the server in the discovery rules, NNMI sees the interface that faces the server as unconnected.

There are two ways to monitor the status of an important interface that connects to an unmonitored node.

- Discover the unmonitored node

When you add an unmonitored node to the NNMi topology, NNMi sees the interfaces connecting the node to the rest of the topology as `CONNECTED`. Then NNMi can poll these interfaces according to the monitoring configuration. NNMi discovers the node as `MANAGED`. Unmanage nodes that you do not want NNMi to monitor.



Each discovered node counts toward the license limit, regardless of whether NNMi is actively managing that node.

- Poll the unconnected interface

You can create a node group containing the network devices that provide connectivity for undiscovered nodes. Then enable polling of unconnected interfaces for the node group.

NNMi polls *all* interfaces on the devices in the node group, which can add a lot of traffic for a device with many interfaces.

Stop Monitoring

The NNMi management modes are used to set devices or interfaces to `UNMANAGED` or `OUT OF SERVICE`. `UNMANAGED` is considered to be a permanent situation; you will never care to know the status of the object. `OUT OF SERVICE` is for temporary situations where one or more objects will be offline and down incidents would be superfluous.

Consider the management mode as an overlay across all group settings. Regardless of its group, polling interval, or type, the State Poller does not communicate with an object when its status is set to `UNMANAGED` or `OUT OF SERVICE`.

Best practice

Some of the devices and/or interfaces you choose to discover and place in the database do not need to be polled. Note those objects which you will permanently set to `UNMANAGED`. You may want to create one or more node groups to enable you to set management modes more easily.

Planning Groups

You must set up node and interface groups before configuring monitoring settings. Therefore, you must consider polling requirements while configuring node and interface groups. Ideally, node and interface groups are configured so that you can monitor important devices frequently, and you can check on non-critical devices less frequently (if at all).

These groups are defined through the **Configuration > Node Groups** or **Configuration > Interface Groups** work spaces and are, by default, the same groups that are used to filter incident, node, interface, and address views. To create a separate set of node or interface filters for configuring monitoring settings, open a node or interface group and select the **Add to View Filter List** check box on the **Node Group** or **Interface Group** form. Click **Save and Close**.

You can set polling types and polling intervals at a node group or interface group level on the **Node Settings** and **Interface Settings** tabs of the **Monitoring Configuration** form.

Determine the criteria by which you want to group interfaces and/or devices by similar polling needs. Here are some factors to consider in your planning:

- Which area of your network contains these devices? Are there timing constraints?
- Do you want to differentiate polling intervals or data gathered by device type? By interface type?
- Does NNMi provide pre-configured groups you can use?

Best practice

You can create group definitions for objects that are likely to go OUT OF SERVICE at the same time, whether by location or some other criteria. For example, you could put all your Cisco routers into OUT OF SERVICE mode while you apply an IOS upgrade.

Interface Groups

Based on your criteria, determine which Interface groups to create. Remember that interface groups are evaluated first (see [Concepts for State Polling](#) on page 71). Interface groups can reference node group membership, so you may end up configuring node groups before interface groups to implement your plan.

Preconfigured interface groups

NNMi has several useful interface groups already configured for you to use. These include:

- All interfaces with an IFType related to ISDN connections
- Interfaces for voice connections
- Interfaces for point-to-point communication
- Software loopback interfaces
- VLAN interfaces
- Interfaces participating in link aggregation protocols

Over time HP may add more default groups to simplify your configuration tasks. You can use existing groups, modify them, or create your own.

Interface groups have two types of qualifiers: node group membership for the hosting node and IFType or other attribute for the interface. You can choose to combine these as follows:

- All interfaces on nodes in a node group are grouped regardless of IFType; do not select any IFTypes or attributes (such as name, alias, description, speed, index, address, or other IFType attributes).
- All interfaces of certain IFTypes or set of attributes are grouped, regardless of the node on which they reside.
- Only interfaces of a certain IFType or attributes that reside on a particular group of nodes are grouped.

Node Groups

After planning interface groups, plan node groups. Not all node groups created for monitoring make sense for filtering views, so you can configure them independently.

Preconfigured node groups

HP provides a default collection of node groups to simplify your configuration tasks. These are based on device categories derived from the system object ID during the Discovery process. The node groups provided by default include:

- Routers
- Networking Infrastructure Devices (such as switches or routers.)
- Microsoft Windows Systems
- Devices for which you do not have the SNMP community string
- Important Nodes. This is used internally by the Causal Engine to provide special handling for devices in the “shadow” of a connector failure. For more information, see *Node Groups As Predefined View Filters* in the NNMi help.

Over time HP may add more default groups to simplify your configuration tasks. You can use existing groups, modify them, or create your own.

You can qualify the definition of related nodes using the following node attributes:

- IP address(es) on the node
- Hostname wildcard convention
- Device Profile derivatives such as category, vendor, and family
- MIB II sysName, sysContact, sysLocation

Best practice

You can create simple, reusable, atomic groups and combine them into hierarchical clusters for monitoring or visualization. Group definitions can overlap, such as “All Routers” and “All systems with IP address ending in .100.” Nodes will probably qualify for multiple groups as well.

Find a balance by creating a rich set of groups for configuration and viewing without overloading the list with superfluous entries that will never be used.

Interaction with Device Profiles

When each device is discovered, NNMi uses its system object ID to index into the list of available Device Profiles. The Device Profile is used to derive additional attributes of the device, such as vendor, product family, and device category.

As you configure node groups, you can use these derived attributes to categorize devices to apply monitoring settings. For example, you may want to poll all switches regardless of vendor throughout your network on a certain polling interval. You can use the derived device category, Switch, as the defining characteristic of your node group. All discovered devices whose system object ID maps to the category, Switches, will receive the configured settings for the node group.

Planning Polling Intervals

For each object group, you select a polling interval that NNMi uses to collect data. The interval can be as short as one minute, or as long as days to best match your Service Level Agreements.

Best practice

Shorter intervals help you become aware of network problems as soon as possible; however, polling too many objects in too short an interval can cause a backlog in the State Poller. Find the best balance between resource utilization and intervals for your environment.

Deciding What Data to Collect

The State Poller service uses polls to gather state information about the monitored devices in your network. Polling can be done using ICMP and/or SNMP.

ICMP (ping)

ICMP address monitoring uses ping requests to verify the availability of each managed IP address.

SNMP

SNMP monitoring verifies that each monitored SNMP agent is responding to SNMP queries.

- The State Poller is highly optimized to collect configured SNMP information from each monitored object with one query at each interval. When you save configuration changes, the State Poller recalculates the group membership of each object and reapplies the configured interval and set of data to collect.
- SNMP monitoring issues SNMP queries for all monitored interfaces and components, requesting the current values from the MIB II interface table, the HostResources MIB, and vendor-specific MIBs. Some values are used for fault monitoring. If you have the NNM iSPI for Metrics installed, some values are used for performance measurement.

SNMP Component Health data

You may enable or disable Component Health monitoring at the global level. Component Health monitoring for faults follows the fault polling interval settings for the device.

Gathering additional data at each poll does not affect the time to execute the poll. However, additional data stored for each object can increase the memory requirements for State Poller.



Performance monitoring settings are only used with the NNM iSPI for Metrics. Component Health monitoring for performance follows the performance polling interval settings for the device.

Best practice

Batching your monitoring configuration changes is less disruptive to State Poller ongoing operation.

Configure State Polling

This section provides configuration tips and provides some configuration examples. After reading the information in this section, see *Configure Monitoring Behavior* in the NNMi help for specific procedures.



It is a good idea to save a copy of the existing configuration before you make any major configuration changes. For more information, see [Best Practice: Save the Existing Configuration](#) on page 38.

Configure Interface Groups and Node Groups

You create interface groups and node groups in the **Configuration** workspace. For more information, see *Creating Groups of Nodes or Interfaces* in the NNMi help.

Examples For example, to configure a node group for ProximiT proxy servers:

- 1 Open **Configuration > Node Groups** and click **New**.
- 2 Name the group **Proxy Servers** and check **Add to View Filter List**.
- 3 On the **Additional Filters** tab, select the **hostname** attribute, and leave the operator set to **=**.
- 4 For value, enter the wildcard as **prox*.example.com**.
If you had configured a device profile and device category for the ProximiT devices, you could use the **Device Filters** tab to access the **Device Category** selector and base the group on the Proxy Server category you created.
- 5 Click **Save and Close** on the group definition.



You must configure node groups before you can reference them in your interface group configuration.

Configure Interface Monitoring

State Poller analyzes interface group membership before node groups. For each of the interface groups you created, as well as any of the preexisting ones you want to use, open the **Monitoring Configuration** dialog and the **Interface Settings** tab to create a custom set of instructions for how State Poller should handle that group. Your instructions will include:

- Enabling or disabling fault polling
- Setting the fault polling interval
- Enabling or disabling performance polling if you have the NNM iSPI for Metrics
- Setting the performance polling interval if you have the NNM iSPI for Metrics
- Setting performance management thresholds if you have the NNM iSPI for Metrics
- Selecting whether NNMi should monitor unconnected interfaces (or unconnected interfaces hosting IP addresses) in the group

You can configure different settings for each interface group. Remember that the State Poller evaluates the list in order from the lowest ordering number to the highest ordering number.

Best practice

Double-check your order numbers, keeping in mind that an object that qualifies for multiple groups has settings applied from the group with the lowest order number.

Configure Node Monitoring

If an object does not qualify for any configured interface group, State Poller evaluates the object for membership in node groups. Settings are applied to the first node group match from the lowest ordering number to the highest ordering number.

For each node group, open the **Monitoring Configuration** form, and then, open the **Node Settings** tab. Create a custom set of instructions as to how State Poller should handle that group. Your instructions can include:

- Enabling or disabling fault polling
- Setting the fault polling interval
- Enabling or disabling performance polling if you have the NNM iSPI for Metrics
- Setting the performance polling interval if you have the NNM iSPI for Metrics
- Setting performance management thresholds if you have the NNM iSPI for Metrics
- Selecting whether NNMi should monitor unconnected interfaces (or unconnected interfaces hosting IP addresses) in the group

You may configure different settings for each node group.

Best practice

Double-check the order numbers, keeping in mind that an object that qualifies for multiple groups has settings applied from the group with the lowest order number.

Verify Default Settings

State Poller applies the settings from the **Default Settings** tab for any object that does not match a defined interface setting or node setting. Review the settings on this tab to ensure they match your environment at the default level. For example, you would rarely poll all unconnected interfaces as a default setting.



Be sure you **Save and Close** all **Monitoring Configuration** dialog boxes all the way back to the console for your changes to be implemented.

Evaluate State Polling

This section lists ways to evaluate the progress and success of the monitoring settings.

Verify the Configuration for Network Monitoring

You can determine the settings that NNMi uses for monitoring a given node or interface, and you can initiate a status poll of a node at any time.

Is the interface or node a member of the right group?

You can verify which interfaces or nodes belong to a group by selecting one of the following in the **Configuration** workspace:

- Node Groups
- Interface Groups

Follow the instructions in the help to show the members of the group. Keep in mind that an object can be a member of multiple groups, and that another group may have a lower ordering number.

Alternatively, you can see the full list of groups to which the object belongs by opening the object (interface or node) and clicking the **Node Groups** or **Interface Groups** tab. This list is alphabetical by group name and does not reflect the ordering numbers that determine which settings are applied.

If the object is not a member of a group:

- 1 Retrieve the device profile for the node in the inventory view.
- 2 Review the attribute mapping for the device profile under **Configuration > Device Profiles**.
- 3 Review the attribute requirements for the node group definition.

If you have a mismatch, you can adjust the category derived in the Device Profile to force that type of device to qualify for your node group. You may need to do an **Actions > Configuration Poll** to update the attributes for the node so that it qualifies.

Which settings are being applied?

To check the monitoring configuration in effect for a specific node, interface, or address, select that object in the appropriate inventory view, and select **Actions > Monitoring Settings**. NNMi displays the current monitoring settings.

Examine the values for **Fault Polling Enabled** and **Fault Polling Interval**. If these values are not as expected, look at the value for **Node Group** or **Interface Group** to see which ordered group match applied.

You may also need to check **Actions > Communication Settings** for the object to ensure traffic has not been disabled for it.

Which data is being collected?

You can initiate a status poll of a specific device to validate that the expected types of polls (SNMP, ICMP) are being performed for that device. Select a node, and then click **Actions > Status Poll**. NNMi performs a real-time status check of the device. The output shows the types and results of the polls being performed. If the types of polls are not what you expect, check the monitoring settings for the node and the respective global, interface, or node settings of the monitoring configuration.

Evaluate the Performance of Status Polling

Evaluate the performance of status polling in your environment by using the information in the state poller health check to quantify and assess the operation of the state poller service.

Is the State Poller keeping up?

At any time, you can check the current health statistics about the state poller service on the **State Poller** tab of the **System Information** window, as described in [Table 2](#).

Table 2 State Poller Health Information

Information	Description
Status	Overall status of the state poller service
Poll counters	<ul style="list-style-type: none">• Collections requested in last minute• Collections completed in last minute• Collections in process
Time to execute skips in last minute	<p>The number of regularly scheduled polls that did not complete within the configured polling interval. A non-zero value indicates that the polling engine is not keeping up or that targets are being polled faster than they can respond.</p> <ul style="list-style-type: none">• What to watch for: If this value continues to increase, there are problems communicating with the target or NNMi is overloaded.
Stale collections in last minute	<p>A stale collection is a collection that has not received a response from the polling engine for at least 10 minutes. A healthy system should never have any stale collections.</p> <ul style="list-style-type: none">• What to watch for: If this value increases consistently, there is a problem with the polling engine.
Poller result queue length	<ul style="list-style-type: none">• What to watch for: This value should be close to 0 most of the time.• Action to take: If this queue size is very large, ovjboss might be running out of memory.
State mapper input queue length	<ul style="list-style-type: none">• What to watch for: This value should be close to 0 most of the time.• Action to take: If this queue size is very large, then check the performance of the NNMi system and the NNMi database.
State updater queue time length	<ul style="list-style-type: none">• What to watch for: This value should be close to 0 most of the time.• Action to take: If this queue size is very large, then check the performance of the NNMi system and the NNMi database.

Tune State Polling

The performance of state polling is affected by the following key variables:

- The number of devices/interfaces to be polled
- The type of polling configured
- The frequency of polling each device

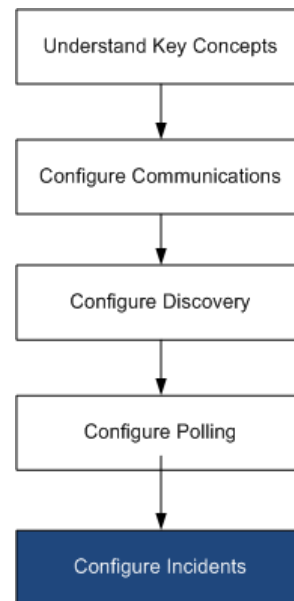
These variables are driven by your network management needs. If you are experiencing performance issues with status polling, consider the following configurations:

- Because polling settings for individual nodes are controlled through their membership in node groups and interface groups, make sure that the groups contain nodes or interfaces with similar polling requirements.
- If you are polling unconnected interfaces or interfaces that host IP addresses, check the configurations to make sure you are only polling the interfaces that are necessary. Enable these polls on the **Node Settings** or **Interface Settings** form (not as a global setting on the **Monitoring Configuration** form) to maintain the most specific control and to select the smallest subset of interfaces to poll.
- Remember that polling unconnected interfaces monitors *all* unconnected interfaces. To monitor only those unconnected interfaces that have IP addresses, enable polling of interfaces that host IP addresses.

Regardless of the monitoring configuration, status polling is dependent on network responsiveness and may be impacted by overall system performance. Although status polling with default polling intervals does not introduce much network load, if the performance of the network link between the server and the polled device is poor, status polling performance is poor. You can configure larger timeouts and a smaller number of retries to reduce the network load, but these configuration changes only go so far. Timely polling requires adequate network performance and sufficient system resources (CPU, memory).

Enabling or disabling the Component Health monitoring has no effect on timeliness of polling. It simply gathers additional MIB objects at the schedule time. However, disabling Component Health monitoring may reduce the amount of memory used by the State Poller.

NNMi Incidents



HP Network Node Manager i Software provides a large number of default incidents and correlations that filter incoming SNMP traps to provide a workable number of incidents in the NNMi console. This chapter provides information to help you fine tune network management by configuring the HP Network Node Manager i Software incidents. This chapter supplements the information in the NNMi help. For an introduction to NNMi incidents and for detailed information about how to configure incidents, see *Configuring Incidents* in the NNMi help.

If you have experience working with NNM 6.x/7.x and you want to understand how event monitoring has changed in NNMi 9.00, see [Customizing Event Monitoring](#) on page 318 for a high-level overview of the differences.

This chapter contains the following topics:

- [Concepts for Incidents](#)
- [Plan Incidents](#)
- [Configure Incidents](#)
- [Evaluate Incidents](#)
- [Tune Incidents](#)

Concepts for Incidents

NNMi collects network status information from the following sources:

- The NNMi Causal Engine analyzes the health of your network and provides the ongoing health status reading for each device. The Causal Engine also extensively evaluates and determines the root cause of network problems whenever possible.
- SNMP traps from network devices. The NNMi Causal Engine uses this information as symptoms during its analysis.
- NNM 6.x/7.x events forwarded from one or more NNM 6.x/7.x management stations.

NNMi converts this network status information into incidents that provide useful information for managing the network. NNMi provides many default incident correlations that reduce the number of incidents for network operators to consider. You can customize the default incident correlations and create new incident correlations to match the network management needs of your environment.

The incident configurations in the NNMi console define the incident types that NNMi can create. If no incident configuration matches a received SNMP trap or NNM 6.x/7.x event, that information is discarded. If the management mode of the trap source is set to NOT MANAGED or OUT OF SERVICE in the NNMi database, NNMi always discards the incoming trap.

Additionally, NNMi discards SNMP traps from network devices that are not in the NNMi topology. For information about changing this default behavior, see *Handle Unresolved Incoming Traps* in the NNMi help.

For more information, see the following:

- *About the Event Pipeline* in the NNMi help
- *The NNMi Causal Engine and Incidents* in the NNMi help
- *NNMi Causal Analysis White Paper*, available from <http://h20230.www2.hp.com/selfsolve/manuals>

Incident Lifecycle

Table 3 describes the stages of an incident's lifecycle.

Table 3 NNMi Incident Lifecycle

Lifecycle State	Description	State Set By	Incident Used By
none	The NNMi event pipeline receives input from all sources and creates incidents as needed.	not applicable	<ul style="list-style-type: none"> • NNMi
Dampened	<p>The incident is in a holding place waiting to be correlated with another incident. The purpose of this waiting period is incident reduction in the incident viewers.</p> <p>The dampening interval can vary per incident type. For more information, see Incident Suppression, Enrichment, and Dampening on page 92.</p>	NNMi	<ul style="list-style-type: none"> • NNMi
Registered	<p>The incident is visible in incident views.</p> <p>The incident is forwarded to any configured destinations (northbound or global manager).</p>	<p>NNMi</p> <p>A user can also set this state in an incident view.</p>	<ul style="list-style-type: none"> • Users • Lifecycle transition actions • Integrations that forward incidents
In Progress	<p>The incident has been assigned to someone who is investigating the problem.</p> <p>The network administrator defines the specific meaning of this state.</p>	User	<ul style="list-style-type: none"> • Users • Lifecycle transition actions • Integrations that forward incidents

Table 3 NNMi Incident Lifecycle (cont'd)

Lifecycle State	Description	State Set By	Incident Used By
Completed	Investigation of the problem indicated by the incident is complete, and a solution is in place. The problem that the incident identifies The network administrator defines the specific meaning of this state.	User	<ul style="list-style-type: none"> • Users • Lifecycle transition actions • Integrations that forward incidents
Closed	Indicates that NNMi determined the problem reported by this Incident is no longer a problem. For example, when you remove an interface from a device, all incidents related to the interface are automatically closed.	User or NNMi	<ul style="list-style-type: none"> • Users • Lifecycle transition actions • Integrations that forward incidents

Trap and Incident Forwarding

Table 4 summarizes the ways to forward traps and incidents from the NNMi management server to another destination. The text following the table compares the NNMi SNMP trap forwarding mechanism with the NNMi northbound interface SNMP trap forwarding mechanism.

Table 4 Supported Ways to Forward Traps and NNMi Incidents

	NNMi Trap Forwarding	NNMi Northbound Interface Trap Forwarding	Global Network Management Trap Forwarding
What to forward	<ul style="list-style-type: none"> • SNMP traps from network devices • NNM 6.x/7.x events from NNM management stations 	<ul style="list-style-type: none"> • SNMP traps from network devices • NNMi management events 	<ul style="list-style-type: none"> • SNMP traps from network devices • NNM 6.x/7.x events from NNM management stations
Forwarding format	SNMPv1, v2c, or v3 traps, as received (SNMPv3 traps can be converted to SNMPv2c traps)	SNMPv2c traps created from NNMi incidents	NNMi incidents
Added information	In most cases, NNMi adds varbinds to identify the original trap source. NNMi does not ever modify SNMPv1 traps.	NNMi adds varbinds to identify the original trap source.	Any information added to the incident by the regional manager processes is retained in the forwarded incident.
Where to configure	Trap Forward Configuration in the Configuration workspace	HPOM, Northbound Interface, or Netcool in the Integration Module Configuration workspace	Forward to Global Managers tab on an SNMP Trap Configuration form or a Remote NNM 6.x/7.x Event Configuration form

Table 4 Supported Ways to Forward Traps and NNMi Incidents (cont'd)

	NNMi Trap Forwarding	NNMi Northbound Interface Trap Forwarding	Global Network Management Trap Forwarding
Notes		<p>NNMi provides several integrations built on the NNMi northbound interface:</p> <ul style="list-style-type: none"> • NNMi Northbound Interface on page 455 • HP NNMi—HPOM Integration (Agent Implementation) on page 487 • HP NNMi Integration Module for Netcool Software on page 525 	Forward the remote incidents that should be visible in the global manager incident views. Forwarded incidents participate in correlations on the global manager.
For more information	<i>Configuring Trap Forwarding</i> in the NNMi help	Using the NNMi Northbound Interface on page 459	<ul style="list-style-type: none"> • <i>Configure Forward to Global Manager Settings for an SNMP Trap Incident</i> in the NNMi help • <i>Configure Forward to Global Managers Settings for a Remote 6.x/7.x Event Incident</i> in the NNMi help

Comparison: Forwarding Third-Party SNMP Traps to Another Application

If you want to forward the SNMP traps that NNMi receives from managed devices to another application, you can use either of the following approaches:

- Use the NNMi SNMP trap forwarding mechanism. For information about how to configure NNMi SNMP trap forwarding, see *Configuring Trap Forwarding* in the NNMi help.
- Use the NNMi northbound interface SNMP trap forwarding mechanism. For information about configuring the NNMi northbound interface to forward received SNMP traps, see [Incidents](#) in [Table 47](#) on page 467.

The approach to trap identification by the receiving application varies with the SNMP trap forwarding mechanism:

- *Windows (all) and UNIX without original trap forwarding*

This description applies to the default and SNMPv3 to SNMPv2c conversion forwarding options.

The NNMi SNMP trap forwarding mechanism on a Windows NNMi management server enriches each SNMP trap before forwarding it to the trap destination. The trap appears to originate from the NNMi management server. (This information also applies to a UNIX NNMi management server for which the original trap forwarding option is not selected on the **Trap Forwarding Destination** form.)

To ensure the correct association between the trap-sending device and the event in the receiving application, the rules for these traps must be customized for the enriched varbinds. Interpret the value from the `originIPAddress` (.1.3.6.1.4.1.11.2.17.2.19.1.1.3) varbind. The `originIPAddress` value is a byte string of generic type `InetAddress`, either `InetAddressIPv4` or `InetAddressIPv6` as determined by the value of `originIPAddressType` (.1.3.6.1.4.1.11.2.17.2.19.1.1.2) varbind. The rule must read the `originIPAddressType` varbind to determine the type of Internet address (`ipv4(1)`, `ipv6(2)`) value in the `originIPAddress` varbind. The rule might also need to convert the `originIPAddress` value to a display string.

For more information about the varbinds that NNMi adds to forwarded traps, see *Trap Varbinds Provided by NNMi* in the NNMi help, RFC 2851, and the following file:

— *Windows*: %NNM_SNMP_MIBS\Vendor\Hewlett-Packard\hp-nnmi.mib

— *UNIX*: \$NNM_SNMP_MIBS/Vendor/Hewlett-Packard/hp-nnmi.mib

- *UNIX with original trap forwarding*

The NNMi SNMP trap forwarding mechanism on a UNIX NNMi management server can forward the traps in the same format as NNMi receives them. Each trap appears as if the managed device sent it directly to the trap destination, so existing trap processing configured in the receiving application should work without modification.

For more information, see the original trap forwarding option in *Trap Forwarding Destination Form* in the NNMi help.

- *NNMi northbound interface (all operating systems)*

The NNMi northbound interface enriches each SNMP trap before forwarding it to the trap destination. The trap appears to originate from the NNMi management server. To ensure the correct association between the trap-sending device and the event in the receiving application, the rules for these traps must be customized for the enriched varbinds. The `IncidentNodeHostname` (1.3.6.1.4.1.11.2.17.19.2.2.21) and `IncidentNodeMgmtAddr` (1.3.6.1.4.1.11.2.17.19.2.2.24) varbinds identify the original trap source.

MIBs

NNMi requires that the following management information base (MIB) files be loaded into the NNMi database:

- All MIB variables used in MIB expressions for the Custom Poller feature, line graphs, or both
- Node components that NNMi monitors for health (for example fan or power supply)
- (NNM iSPI for Metrics) All MIB variables used in threshold monitoring

NNMi requires that the following management information base (MIB) files, or the traps defined in the MIB files, be loaded into the NNMi database:

- All SNMP traps that you want to forward to a northbound destination
- (NNM iSPI NET) All MIB variables accessed from Trap Analytics reports

Custom Incident Attributes

NNMi uses custom incident attributes (CIAs) to attach additional information to incidents.

- For an SNMP trap incident, NNMi stores the original trap varbinds as CIAs for the incident.
- For a management event incident, NNMi adds pertinent information (for example, com.hp.ov.nms.apa.symptom) as CIAs for the incident.

You can use incident CIAs to narrow the scope of configurations such as incident lifecycle transition actions, suppression, deduplication, and enrichment. You can also use CIAs to narrow the availability of the menu items on the Actions menu for an incident view or form.

To determine which CIAs NNMi adds for any given incident, open a sample incident from an incident view, and look at the information on the Custom Attributes tab.

CIAs Added to Closed Management Event Incidents

When the NNMi Causal Engine determines that the conditions that caused a management event incident no longer apply, NNMi sets that incident's lifecycle state to CLOSED and adds the CIAs listed in [Table 5](#) to the incident. NNMi console users can see this information in the **Correlation Notes** field of the **Incident** form. Lifecycle transition actions can use the values of the CIAs directly.

Table 5 Custom Incident Attributes for a Closed Incident

Name	Description
cia.reasonClosed	The reason that NNMi cancelled or closed the incident. This reason is also the conclusion name, for example NodeUp or InterfaceUp. If this field is not set, an NNMi console user closed the incident. To determine the NNMi expected values of the cia.reasonClosed CIA, see <i>How NNMi Closes Incidents</i> in the NNMi help.
cia.incidentDurationMs	The duration, in milliseconds, of the outage, as measured by NNMi from when the status goes down and comes back up. This value is the difference of the cia.timeIncidentDetectedMs and cia.timeIncidentResolvedMs CIAs. It is a more accurate measurement than comparing the timestamps of down and up incidents.
cia.timeIncidentDetectedMs	The timestamp, in milliseconds, when the NNMi Causal Engine first detected the problem.
cia.timeIncidentResolvedMs	The timestamp, in milliseconds, when the NNMi Causal Engine detected that the problem has been resolved.

NNMi adds the CIAs listed in [Table 5](#) to most primary and secondary root cause incidents. For example, a NodeDown incident can have InterfaceDown and AddressDown incidents as secondary root causes. When NNMi closes the NodeDown incident, NNMi also closes the secondary incidents and adds the CIAs with values for each incident context to the secondary incidents.

NNMi does not add the CIAs listed in [Table 5](#) to the following default management event incident types:

- Incidents that an NNMi console user closes manually
- Incidents that NNMi closes in response to an object being deleted from the NNMi database
- IslandGroupDown incidents
- NnmClusterFailover, NnmClusterLostStandby, NnmClusterStartup, and NnmClusterTransfer incidents
- Incidents in the following families:
 - Correlation
 - License
 - NNMi Health
 - Trap Analysis

Incident Reduction

NNMi provides the following customizable correlations for reducing the number of incidents that network operators see in the NNMi console:

- Pairwise correlation—One incident cancels another incident.
- Deduplication correlation—When multiple copies of an incident are received within the specified time window, correlate the duplicates under a deduplication incident. The time window restarts for each newly received duplicate incident. In this way, NNMi correlates the duplicate incidents until it has not received any duplicates for the entire duration of the correlation time window.
- Rate correlation—When the specified number of copies on an incident are received within the specified time window, correlate the duplications under a rate incident. NNMi generates the rate incident when the specified number of incidents has been received, regardless of how much time remains in the time window.

Incident Suppression, Enrichment, and Dampening

NNMi provides a rich feature set for getting the most value from incidents. For each incident type, you can specifically define when an incident is of interest with the following incident configuration options:

- **Suppression**—When an incident matches the suppression configuration, that incident does not appear in the NNMi console incident views. Incident suppression is useful for incidents (for example, SNMPLinkDown traps) that are important for some nodes (for example routers and switches) but not others.
- **Enrichment**—When an incident matches the enrichment configuration, NNMi changes one or more incident values (for example, severity or message) according to the contents of the incident. Incident enrichment is useful for processing traps (for example, RMONFallingAlarm) that carry the distinguishing information in the trap varbinds (payload).
- **Dampening**—When an incident matches the dampening configuration, NNMi delays activity for that incident for the duration of the dampen interval. Incident dampening provides time for the NNMi Causal Engine to perform root cause analysis on the incident, which is useful for providing fewer, more meaningful incidents in the NNMi console.

For each incident type NNMi provides the following levels of configuration for suppression, enrichment, and dampening:

- **Interface group settings**—Specify incident behavior when the source object is a member of an NNMi interface group. You can specify different behavior for each interface group.
- **Node group settings**—Specify incident behavior when the source object is a member of an NNMi node group. You can specify different behavior for each node group.
- **Default settings**—Specify default incident behavior.

For each incident configuration area (suppression, enrichment, and dampening), NNMi uses the following procedure to determine the behavior of a specific incident:

- 1 Check the interface group settings:
 - If the source object matches any interface group settings, carry out the behavior defined in the match with the lowest ordering number and stop looking for a match.
 - If the source object does not match any interface group settings, continue with [step 2](#).
- 2 Check the node group settings:
 - If the source object matches any node group settings, carry out the behavior defined in the match with the lowest ordering number and stop looking for a match.
 - If the source object does not match any node group settings, continue with [step 3](#).
- 3 Carry out the behavior defined in the default settings, if any.

Lifecycle Transition Actions

A lifecycle transition action is an administrator-provided command that runs when an incident lifecycle state changes to match the action configuration. An incident action configuration is specific to one lifecycle state for one incident type. The action configuration identifies the command to run when this incident type transitions to the specified lifecycle state. The command can include arguments that pass incident information to the action code.

The action code can be any Jython file, script, or executable that runs correctly on the NNMi management server. The action code can be specific to one incident type, or it can process many incident types. For example, you might create action code that pages a network operator when NNMi creates a ConnectionDown, NodeDown, or NodeOrConnectionDown incident. You would configure three incident actions, one for the REGISTERED lifecycle state for each of these incident types.

Similarly, the action code can be specific to one lifecycle state change, or it can respond to several lifecycle state changes. For example, you might create action code that generates a trouble ticket when NNMi creates an InterfaceDown incident and closes the trouble ticket when the InterfaceDown incident is canceled. You would configure two incident actions for the InterfaceDown incident, one for the REGISTERED state and one for the CLOSED state.

Each action configuration can include a payload filter based on CIAs that limits when the action is run. For additional filtering, you can use incident enrichment to add a CIA to the incident. NNMi determines the value of that attribute from the incident source. For example, if you have added a custom attribute to some nodes, you can add this information to the incident as a CIA and then base the payload filter for an incident action on this attribute value.

Plan Incidents

Make decisions in the following areas:

- Which Device Traps Should NNMi Process?
- Which Incidents Should NNMi Display?
- How Should NNMi Respond to Incidents?
- Should NNMi Receive Traps from an NNM Management Station?
- Should NNMi Forward Traps to Another Event Receiver?

Which Device Traps Should NNMi Process?

Identify the device traps that are of interest in your network, and plan an incident configuration for each trap. NNMi can process traps without the MIB being loaded into NNMi. If the MIB contains TRAP-TYPE or NOTIFICATION-TYPE macros, you can create skeleton incident configurations for the traps defined in the MIB.

Decide whether you want to see traps from devices that are not in the NNMi topology.

Which Incidents Should NNMi Display?

The default set of incidents is a good place to start. You can expand and reduce the incident set over time.

Plan which incidents can be reduced through deduplication, rate configuration, and pairwise correlation.

How Should NNMi Respond to Incidents?

What actions (for example, sending an email message to a network operator) should NNMi take when certain incidents occur? At what lifecycle state should each action run?

Should NNMi Receive Traps from an NNM Management Station?

If your environment includes one or more NNM 6.x/7.x management stations that will continue managing areas of the network in conjunction with NNMi, identify the NNM 6.x/7.x events that will help NNMi operators manage the network. Plan an incident configuration for each NNM 6.x/7.x event that should be available in the NNMi console.

Should NNMi Forward Traps to Another Event Receiver?

If your environment includes a third-party trap consolidator, decide whether to use the NNMi SNMP trap forwarding mechanism with the NNMi northbound interface SNMP trap forwarding mechanism.

If you choose the NNMi northbound interface SNMP trap forwarding mechanism, load the MIBs for all traps that NNMi will forward to the event receiver.

Configure Incidents

This section lists configuration tips and provides some configuration examples. After reading the information in this section, see *Configuring Incidents* in the NNMi help for specific procedures.



It is a good idea to save a copy of the existing configuration before you make any major configuration changes. For more information, see [Best Practice: Save the Existing Configuration](#) on page 38.

- Configure the incident types that you planned. If possible, start with the skeleton incident configurations from the traps defined in the MIB.
- Load any MIBs that are required for trap forwarding.
- Verify that devices are configured to send traps to the NNMi management server.

Configuring Incident Suppression, Enrichment, and Dampening

While configuring incident suppression, enrichment, and dampening, note the following:

- For each interface group, node group, or default setting, you can specify a payload filter that further refines when the configuration is applicable.
- Configure interface group settings on the **Interface Settings** tab of an incident configuration form.
- Configure node group settings on the **Node Settings** tab of an incident configuration form.
- Configure default settings on the **Suppression**, **Enrichment**, and **Dampening** tabs of an incident configuration form.

Configuring Lifecycle Transition Actions

While configuring lifecycle transition actions, note the following:

- By default, NNMi runs actions in the following location:

— *Windows*: %NnmDataDir%\shared\nnm\actions

— *UNIX*: \$NNM_DATA/shared/nnm/actions

If an action is not in this location, specify the absolute path to the action in the **Command** field of the **Lifecycle Transition Action** form.



Jython files must be placed in the `actions` directory.

- Each time you make a change to the action configuration, NNMi rereads the actions directory for Jython files and loads them into NNMi.
- Actions are enabled as a group for an incident type.
- For information about the NNMi information that you can pass to an action, see *Valid Parameters for Configuring Incident Actions* in the NNMi help.

Evaluate Incidents

This section lists ways to evaluate the incident configuration.

- Verify that NNMi receives traps from all managed devices in the network.

If NNMi is not receiving traps, verify the configuration of the firewall on the NNMi management server.



Some anti-virus software includes a firewall that is configured separately from the system firewall.

- Verify that the most important traps are converted to incidents.

- Verify that incident actions run at the correct lifecycle state transitions.
- Verify that NNMi is handling incidents as expected.

The **Actions > Incident Configuration Reports** menu contains several options for testing an existing incident against the current configuration of that incident type. Using one of these menu items does not change the incidents currently in the NNMi console.

Tune Incidents

Reduce the number of incidents in the NNMi console incident views. Use any of the following methods:

- Disable the incident configuration for any incident types that are not needed in the NNMi console.
- Set the management mode of the network objects that you do not need to monitor to **NOT MANAGED** or **OUT OF SERVICE**. NNMi discards any incoming traps from these nodes and their interfaces.
- Set NNMi to not monitor some network objects. NNMi discards any incoming traps from trap sources that are not monitored.
- Identify additional criteria for or relationships between incoming incidents. When these criteria or relationships occur, NNMi modifies the flow of incidents by recognizing the criteria or patterns of incoming management events or SNMP traps and nesting related incidents as correlated children.

Advanced Configuration

This section contains the following chapters:

- Working with Certificates for NNMi
- Using Single Sign-On with NNMi
- Integrating NNMi with a Directory Service through LDAP
- Global Network Management
- Configuring NNMi for Application Failover
- Configuring NNMi in a High Availability Cluster
- Configuring NNMi Advanced for IPv6

Working with Certificates for NNMi

A certificate identifies the web server to the browser. This certificate can be self-signed or CA (Certificate Authority) signed. The `nmn.keystore` file stores private keys and certificates with their corresponding public keys. The `nmn.truststore` file contains certificates from other parties that you expect to communicate with, or from Certificate Authorities that you trust to identify other parties. NNMi includes a self-signed certificate in both of the `nmn.keystore` and `nmn.truststore` files.

To use certain NNMi features, NNMi management servers need to share their certificates with one another. This chapter contains configuration instructions for copying these certificates among NNMi management servers and using the `nmncertmerge.ovpl` script to merge these certificates into the `nmn.keystore` and `nmn.truststore` files.

This chapter contains the following topics:

- [Putting it All Together](#)
- [Generating a Certificate Authority Certificate](#)
- [Configuring Application Failover to use Self-Signed Certificates](#)
- [Configuring Application Failover to use a Certificate Authority](#)
- [Configuring High Availability to use Self-Signed or Certificate Authority Certificates](#)
- [Configuring the Global Network Management Feature to use Self-Signed Certificates](#)
- [Configuring the Global Network Management Feature to use a Certificate Authority](#)
- [Configuring Global Network Management with Application Failover to use Self-Signed Certificates](#)
- [Configuring an SSL Connection to the Directory Service](#)

Putting it All Together

Use the following information to guide you in configuring certificates for your special needs:

- If you are using CA certificates, follow the instructions shown in [Generating a Certificate Authority Certificate](#) on page 101.
- If you configured your global and/or regional NNMi management servers to use the application failover feature there are some additional configuration steps. Merge the NNMi management servers' `nmn.keystore` and `nmn.truststore` files for each cluster before completing the global network management configuration, as described in the [Configuring Application Failover to use Self-Signed Certificates](#) on page 103.
- If you need to use a Certificate Authority, and you configured your global and/or regional NNMi management servers to use the application failover feature, there are some additional configuration steps. First, follow the instructions shown in [Generating a Certificate Authority Certificate](#) on page 101; then merge the NNMi management servers' `nmn.keystore` and `nmn.truststore` files for each cluster before completing the global network management configuration, as described in the [Configuring Application Failover to use a Certificate Authority](#) on page 104.
- If you configured your global and/or regional NNMi management servers to use High Availability, create the self-signed certificate in the `nmn.keystore` and `nmn.truststore` files for the virtual hostname before completing the global network management configuration, as described in [Configuring High Availability to use Self-Signed or Certificate Authority Certificates](#) on page 105.
- After you have each HA or application failover cluster properly configured, enable the global network management feature by copying the `nmn.truststore` file from the active regional node to the active global node, then merging the truststore. You must do this for each active regional node. Review the information shown in [Configuring Global Network Management with Application Failover to use Self-Signed Certificates](#) on page 109. If the NNMi management servers use CA certificates generated using the procedure shown in [Generating a Certificate Authority Certificate](#) on page 101, then those CA certificates are the only certificates you need to merge into the global truststore.
- If you configure your NNMi management servers in a global network management configuration, then decide later to change the regional and/or global to be in an application failover cluster, follow the instructions shown in [Configuring Application Failover to use Self-Signed Certificates](#) on page 103. You need to use the commands shown in that section to configure your `nmn.keystore` and `nmn.truststore` files correctly; then copy the modified `nmn.truststore` file to the global NNMi management server and merge it into its `nmn.truststore` file.
- If you configure your NNMi management servers in a global network management configuration, then decide later to change the regional and/or global to use HA, follow the instructions shown in [Configuring High Availability to use Self-Signed or Certificate Authority Certificates](#) on page 105.
- After directory service communications are enabled, NNMi uses the LDAP protocol for retrieving data from a directory service. If the directory service requires an SSL connection, follow the instructions show in [Configuring an SSL Connection to the Directory Service](#) on page 110.

Generating a Certificate Authority Certificate

If you plan to use a CA (Certificate Authority), complete the following steps to generate a CA certificate.

▶ If you plan to use a CA with NNMI, sign the certificate using the RSA algorithm. The DSA algorithm is not supported.

1 Change to the directory on the NNMI management server that contains the `nnm.keystore` and `nnm.truststore` files:

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

2 Save a backup copy of the `nnm.keystore` file.

3 Generate a private key from your system. Use the `keytool` command to generate this private key:

a Run the following command:

- *Windows:* `%NnmInstallDir%\nonOV\jdk\nnm\bin\keytool.exe -genkeypair -validity 3650 -keyalg rsa -keystore nnm.keystore -storepass nnmkeypass -alias myserver.mydomain`
- *UNIX:* `$NnmInstallDir/nonOV/jdk/nnm/bin/keytool.exe -genkeypair -validity 3650 -keyalg rsa -keystore nnm.keystore -storepass nnmkeypass -alias myserver.mydomain`

▶ The alias, referred to as `myserver.mydomain` in this example, identifies this newly-created key. Although the alias can be any string, HP recommends you use the fully-qualified domain name of your system for the `myserver.mydomain` alias variable.

b Enter the requested information.

▶ When prompted for your first and last name, enter the fully-qualified domain name of your system.

4 Run the following command to create a CSR (Certificate Signing Request) file:

- *Windows:* `$NnmInstallDir/nonOV/jdk/nnm/bin/keytool.exe -keystore nnm.keystore -certreq -storepass nnmkeypass -alias myserver.mydomain -file CERTREQFILE`
- *UNIX:* `/opt/OV/nonOV/jdk/nnm/bin/keytool -keystore nnm.keystore -certreq -storepass nnmkeypass -alias myserver.mydomain -file CERTREQFILE .`

▶ For more information about the `keytool` command, search for “Key and Certificate Management Tool” at java.sun.com.

5 Send the CSR to your CA signing authority. They should provide you with a signed certificate, referred to as `myserver.crt`. The `myserver.crt` file contains both the server certificate and the CA (Certified Authority) certificate. Use the CA certificate when importing the server certificate.

- 6 Copy the files containing these certificates to a location on the NNMi management server. For this example, copy the files to the following location:
 - *Windows:* %NNM_DATA%\shared\nnm\certificates
 - *UNIX:* \$NNM_DATA/shared/nnm/certificates

Use the certificates you generated in the previous steps to replace the self-signed certificate:

- 1 Change to the directory on the NNMi management server that contains the `nnm.keystore` and `nnm.truststore` files:
 - *Windows:* %NNM_DATA%\shared\nnm\certificates
 - *UNIX:* \$NNM_DATA/shared/nnm/certificates
- 2 Run the following command to import the signed certificate and the CA certificate into the NNMi `nnm.keystore` file:

Windows:

- `%NnmInstallDir%\nonOV\jdk\nnm\bin\keytool -importcert -trustcacerts -keystore nnm.keystore -storepass nnmkeypass -alias myserver.mydomain -file myserver.crt`

UNIX:

- `$NnmInstallDir/nonOV/jdk/nnm/bin/keytool -importcert -trustcacerts -keystore nnm.keystore -storepass nnmkeypass -alias myserver.mydomain -file myserver.crt`



If you use the `-storepass` option and provide the password, the keystore program does not prompt you for the keystore password. If you do not use the `-storepass` option, enter `nnmkeypass` when prompted for the keystore password.

- 3 When prompted to trust the certificate, enter: **y**

The output from this command is of the form:

```
Owner: CN=NNMi_server.example.com
Issuer: CN=NNMi_server.example.com
Serial number: 494440748e5
Valid from: Tue Oct 28 10:16:21 MST 2008 until: Thu Oct 04
11:16:21 MDT 2108
Certificate fingerprints:
MD5: 29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02
SHA1: C4:03:7E:C4:03:7E:C4:03:7E:C4:03:7E:C4:03:7E:C4:03
Trust this certificate? [no]: y
Certificate was added to keystore
```

- 4 Run the following commands to import the CA certificate into the NNMi `nnm.truststore` file:

— *Windows:*

```
%NnmInstallDir%\nonOV\jdk\nnm\bin\keytool -import -alias myca -keystore nnm.truststore -file myca.crt
```

— *UNIX:*

```
$NnmInstallDir/nonOV/jdk/nnm/bin/keytool -import -alias myca -keystore nnm.truststore -file myca.crt
```

- 5 When prompted for the truststore password, enter: **ovpass**.

Example output for importing a certificate into the keystore

- 6 Examine the contents of the trust store:
 - *Windows:*
`%NnmInstallDir%\nonOV\jdk\nnm\bin\keytool -list \-keystore nnm.truststore`
 - *UNIX:*
`$NnmInstallDir/nonOV/jdk/nnm/bin/keytool -list \-keystore nnm.truststore`

When prompted for the truststore password, enter: **ovpass**

Example trust store output

The trust store output is of the form:

```
Keystore type: jks
Keystore provider: SUN
Your keystore contains 1 entry
nnmi_ldap, Nov 14, 2008, trustedCertEntry,
Certificate fingerprint (MD5):
29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02
```



The trust store can include multiple certificates.

- 7 Edit the following file:
 - *Windows:* `%NNM_CONF%\nnm\props\nms-local.properties`
 - *UNIX:* `$NNM_CONF/nnm/props/nms-local.properties`
- 8 Update the `com.hp.ov.nms.ssl.KEY_ALIAS` variable to the value you used for `myserver.mydomain`. Make sure to save your work.
- 9 Restart `ovjboss` by running these commands:
 - a **ovstop ovjboss**
 - b **ovstart ovjboss**

Configuring Application Failover to use Self-Signed Certificates

Using Self-Signed Certificates with Application Failover



When configuring the application failover feature, you must merge the `nnm.keystore` and `nnm.truststore` file content for both nodes into a single `nnm.keystore` and `nnm.truststore` file. Complete the following steps to configure the application failover feature to use self-signed certificates based on the above diagram.

- 1 Change to the following directory on Server Y before completing [step 2](#):
 - *Windows:* `%NNM_DATA%\shared\nnm\certificates`
 - *UNIX:* `$NNM_DATA/shared/nnm/certificates`

- 2 Copy the `nmm.keystore` and `nmm.truststore` files from Server Y to some temporary location on Server X. The remaining steps refer to these file locations as `<keystore>` and `<truststore>`.
- 3 Run the following command on Server X to merge Server Y's certificates into Server X's `nmm.keystore` and `nmm.truststore` files.

Windows:

```
nmmcertmerge.ovpl -keystore <keystore> -truststore
<truststore>
```

UNIX:

```
nmmcertmerge.ovpl -keystore <keystore> -truststore
<truststore>
```

- 4 Copy the merged `nmm.keystore` and `nmm.truststore` files from server X to server Y, so that both nodes have the merged files. The location of these files is as follows:
 - *Windows:* %NNM_DATA%\shared\nnm\certificates
 - *UNIX:* \$NNM_DATA/shared/nnm/certificates
- 5 Continue configuring the application failover feature at [step 6 on page 182](#).



After you start the application failover feature, NNMi automatically replicates the merged keystore and truststore information from NNMi_active to NNMi_standby.

Configuring Application Failover to use a Certificate Authority

Using CA Certificates with Application Failover



When configuring the application failover feature, you must merge the `nmm.keystore` and `nmm.truststore` file content for both nodes into a single `nmm.keystore` and `nmm.truststore` file. Complete the following steps to configure the application failover feature to use CA certificates based on the above diagram.

- 1 Follow the instructions shown in [Generating a Certificate Authority Certificate](#) on page 101 for NNMi_standby.
- 2 Change to the following directory on NNMi_standby before completing [step 3](#):
 - *Windows:* %NNM_DATA%\shared\nnm\certificates
 - *UNIX:* \$NNM_DATA/shared/nnm/certificates
- 3 Copy the `nmm.keystore` and `nmm.truststore` files from Server Y to some temporary location on Server X. The remaining steps refer to these file locations as `<keystore>` and `<truststore>`.

- 4 Run the following command on Server X to merge Server Y's certificates into Server X's `nmm.keystore` and `nmm.truststore` files.

Windows:

```
nmmcertmerge.ovpl -keystore <keystore> -truststore  
<truststore>
```

UNIX:

```
nmmcertmerge.ovpl -keystore <keystore> -truststore  
<truststore>
```

- 5 Copy the merged `nmm.keystore` and `nmm.truststore` files from server X to server Y, so that both nodes have the merged files. The location of these files is as follows:
 - *Windows:* %NNM_DATA%\shared\nmm\certificates
 - *UNIX:* \$NNM_DATA/shared/nmm/certificates
- 6 Continue configuring the application failover feature at [step 6 on page 182](#).



After you start the application failover feature, NNMi automatically replicates the merged keystore and truststore information from Server X to Server Y

Configuring High Availability to use Self-Signed or Certificate Authority Certificates

Using Certificates with HA



Configuring High Availability to use Self-Signed Certificates

The process for configuring NNMi for HA correctly shares the self-signed certificate among the primary and secondary cluster nodes. You do not need to take any extra steps to use the default certificate with NNMi running under HA.

Configuring High Availability for a New Certificate

Suppose you create a new self-signed or CA certificate, referred to as `newcert`. Complete the following steps to configure HA with this new CA or self-signed certificate.



You can complete this procedure before or after configuring NNMi for HA, as described in [Configuring HA](#) on page 209.

- 1 Change to the following directory on NNMi_HA1 before completing [step 2](#):
 - *Windows*: %NNM_DATA%\shared\nnm\certificates
 - *UNIX*: \$NNM_DATA/shared/nnm/certificates
- 2 On NNMi_HA1, run the following commands to import newcert into the nnm.keystore file:
 - *Windows*: %NnmInstallDir%\nonOV\jdk\nnm\bin\keytool -import -alias newcert_Alias -keystore nnm.keystore -file newcert
 - *UNIX*: \$NnmInstallDir/nonOV/jdk/nnm/bin/keytool -import -alias newcert_Alias -keystore nnm.keystore -file newcert
- 3 Edit the following file on both the active (NNMi_HA1) and the standby (NNMi_HA2) nodes:
 - *Windows*: %NNM_DATA%\conf\nnm\props\nms-local.properties
 - *UNIX*: \$NNM_DATA/conf/nnm/props/nms-local.properties
- 4 Change the following line in the nms-local.properties file on both NNMi_HA1 and NNMi_HA2.

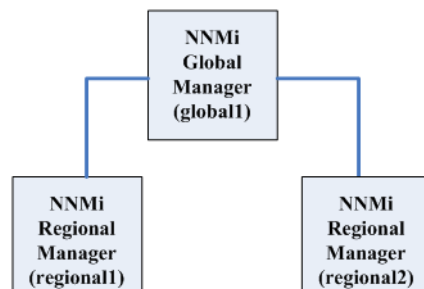

```
com.hp.ov.nms.ssl.KEY_ALIAS = newcert
```
- 5 Save your changes.

Configuring the Global Network Management Feature to use Self-Signed Certificates

During NNMi installation, the installation script creates a self-signed certificate for the NNMi management server. This certificate contains an alias that includes the fully-qualified domain name of the node. The installation script adds this self-signed certificate to the NNMi management server's nnm.keystore and nnm.truststore files.

Suppose you want your global network management configuration to model the following diagram:

Using Self-signed Certificates with Global Network Management



Complete the following steps to configure the global network management feature to use self-signed certificates based on the above diagram.

- 1 Change to the following directory on `regional1` and `regional2` before completing [step 2](#):
 - *Windows*: `%NNM_DATA%\shared\nnm\certificates`
 - *UNIX*: `$NNM_DATA/shared/nnm/certificates`
- 2 Copy the `nmn.truststore` files from the above locations on `regional1` and `regional2` to some location on `global1`.
- 3 Run the following command on `global1` to merge the `regional1` and `regional2` certificates into `global1`'s `nmn.truststore` file.

Windows:

- a `nmncertmerge.ovpl -truststore regional1_nmn.truststore_location`
- b `nmncertmerge.ovpl -truststore regional2_nmn.truststore_location`

UNIX

- a `nmncertmerge.ovpl -truststore regional1_nmn.truststore_location`
- b `nmncertmerge.ovpl -truststore regional2_nmn.truststore_location`

- 4 Run the following command sequence on `global1`:

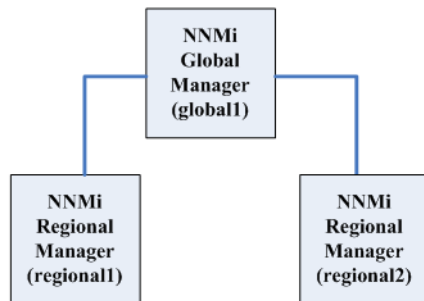
- a `ovstop ovjboss`
- b `ovstart ovjboss`

Configuring the Global Network Management Feature to use a Certificate Authority

During NNMi installation, the installation script creates a self-signed certificate for the NNMi management server. This certificate contains an alias that includes the fully-qualified domain name of the node. The installation script adds this self-signed certificate to the NNMi management server's `nmn.keystore` and `nmn.truststore` files.

Suppose you want your global network management configuration to model the following diagram:

Using CA Certificates with Global Network Management



- 1 Follow the instructions shown in [Generating a Certificate Authority Certificate](#) on page 101 for regional1 and regional2.
- 2 Change to the following directory on regional1 and regional2 before completing [step 3](#).
 - *Windows*: %NNM_DATA%\shared\nnm\certificates
 - *UNIX*: \$NNM_DATA/shared/nnm/certificates
- 3 Copy the nnm.truststore files from the above locations on regional1 and regional2 to some location on global1.
- 4 Run the following command on global1 to merge the regional1 and regional2 certificates into global1's nnm.truststore file.

Windows:

- a `nnmcertmerge.ovpl -truststore regional1_nnm.truststore_location`
- b `nnmcertmerge.ovpl -truststore regional2_nnm.truststore_location`

UNIX

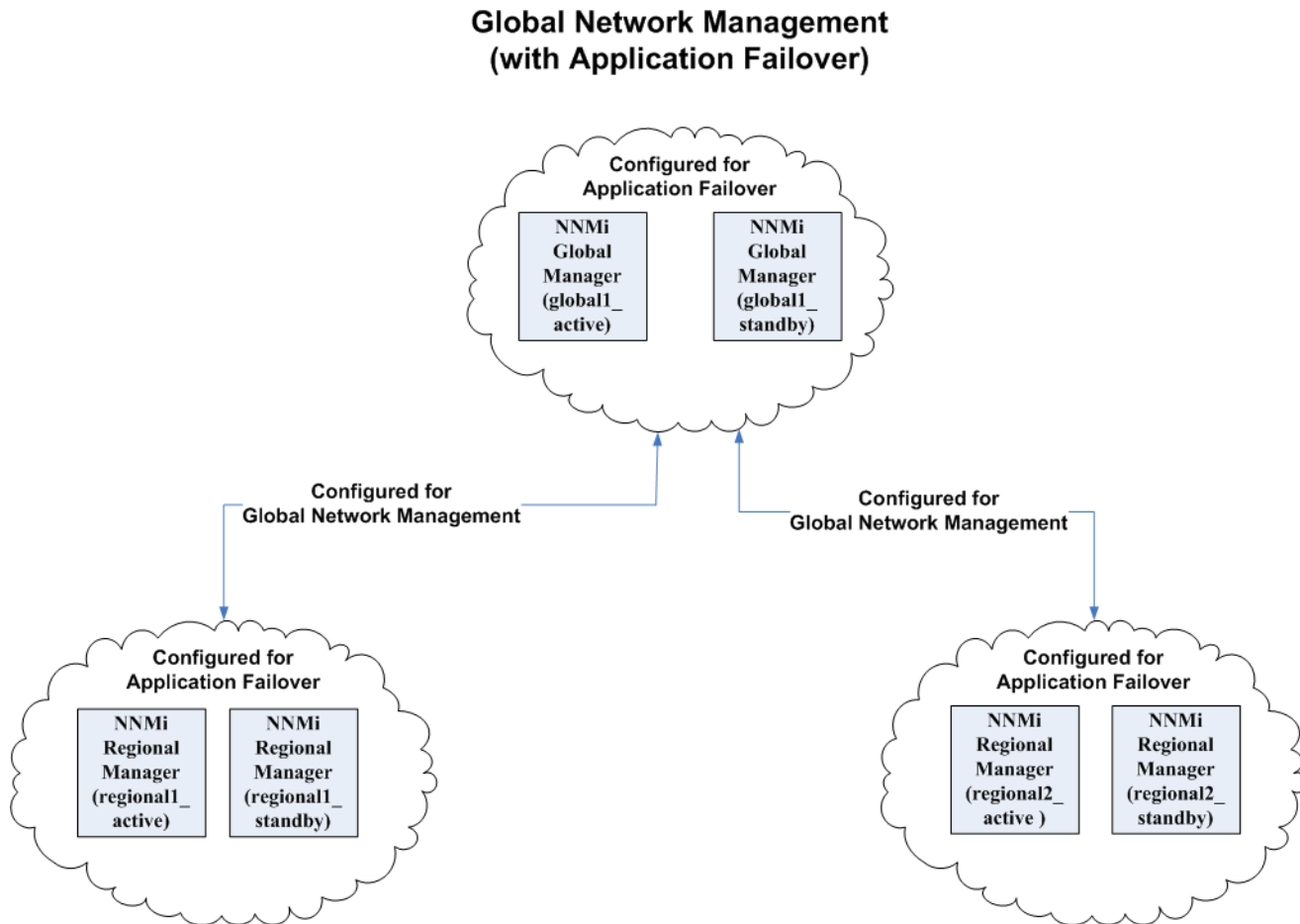
- a `nnmcertmerge.ovpl -truststore regional1_nnm.truststore_location`
- b `nnmcertmerge.ovpl -truststore regional2_nnm.truststore_location`

- 5 Run the following command sequence on global1:
 - a `ovstop ovjboss`
 - b `ovstart ovjboss`

Configuring Global Network Management with Application Failover to use Self-Signed Certificates

As discussed above, during NNMI installation the installation script creates a self-signed certificate for the NNMI management server. This certificate contains an alias that includes the fully-qualified domain name of the node. The installation script adds this self-signed certificate to the NNMI management server's `nmm.keystore` and `nmm.truststore` files.

Suppose you want your global network management configuration to model the application failover feature as shown in the following diagram:



Complete the following steps to configure the global network management feature to work with application failover based on the above diagram:

- 1 Follow the instructions shown in [Configuring Application Failover to use Self-Signed Certificates](#) on page 103 for each application failover cluster shown in the above diagram.
- 2 Complete the configuration for application failover shown in [Application Failover Basic Setup](#) on page 180.

- 3 Follow the instructions shown in [Configuring the Global Network Management Feature to use Self-Signed Certificates](#) on page 106 for `regionall_active` and `regional2_active`.

Configuring an SSL Connection to the Directory Service

By default, when directory service communications are enabled, NNMi uses the LDAP protocol for retrieving data from a directory service. If your directory service requires an SSL connection, you must enable the SSL protocol to encrypt the data that flows between NNMi and the directory service.

SSL requires a trust relationship between the directory service host and the NNMi management server. To create this trust relationship, add a certificate to the NNMi trust store. The certificate confirms the identity of the directory service host to the NNMi management server.

To install a trust store certificate for SSL communications, follow these steps:

- 1 Obtain your company's trust store certificate from the directory server. The directory service administrator should be able to give you a copy of this text file.
- 2 Change to the directory that contains the NNMi trust store:
 - *Windows:* `%NNM_DATA%\shared\nnm\certificates`
 - *UNIX:* `$NNM_DATA/shared/nnm/certificates`

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

- 3 Import your company's trust store certificate into the NNMi trust store:
 - a Run the following command:

```
— Windows:
%NnmInstallDir%\nonOV\jdk\b\bin\keytool -import \
-alias nnmi_ldap -keystore nnm.truststore \
-file <Directory_Server_Certificate.txt>

— UNIX:
$NnmInstallDir/nonOV/jdk/b/bin/keytool -import \
-alias nnmi_ldap -keystore nnm.truststore \
-file <Directory_Server_Certificate.txt>
```

Where `<Directory_Server_Certificate.txt>` is your company's trust store certificate.

- b When prompted for the keystore password, enter: **ovpass**
- c When prompted to trust the certificate, enter: **y**

The output from this command is of the form:

```
Owner: CN=NNMi_server.example.com
Issuer: CN=NNMi_server.example.com
Serial number: 494440748e5
Valid from: Tue Oct 28 10:16:21 MST 2008 until: Thu Oct 04
11:16:21 MDT 2108
Certificate fingerprints:
MD5: 29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02
```

Example output for
importing a
certificate into the
trust store

```
SHA1: C4:03:7E:C4:03:7E:C4:03:7E:C4:03:7E:C4:03:7E:C4:03
Trust this certificate? [no]: y
Certificate was added to keystore
```

4 Examine the contents of the trust store:

- *Windows:*
`%NnmInstallDir%\nonOV\jdk\b\bin\keytool -list \
-keystore nnm.truststore`
- *UNIX:*
`$NnmInstallDir/nonOV/jdk/b/bin/keytool -list \
-keystore nnm.truststore`

When prompted for the keystore password, enter: **ovpass**

Example trust store
output

The trust store output is of the form:

```
Keystore type: jks
Keystore provider: SUN
Your keystore contains 1 entry
nnmi_ldap, Nov 14, 2008, trustedCertEntry,
Certificate fingerprint (MD5):
29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02
```



The trust store can include multiple certificates.

5 Restart ovjboss by running these commands:

- a **ovstop ovjboss**
- b **ovstart ovjboss**

For more information about the `keytool` command, search for “Key and Certificate Management Tool” at java.sun.com.

Using Single Sign-On with NNMi

You can configure HP Network Node Manager i Software single sign-on (SSO) to facilitate access to NNM iSPIs from the NNMi console. With SSO, when you sign in to the NNMi console, you receive access to other NNM iSPIs without needing to sign in again. SSO provides easier access to NNM iSPIs while maintaining a secure level of NNM iSPI access. After you sign out of the NNMi console, you must re-enter your sign-in credentials to access NNM iSPI URLs outside the NNMi console.

SSO Access for NNMi

To browse among several NNMi management servers, you need to do one of the following:

- Edit the `lwssofmconf.xml` file and make the `initString` and `domain` parameters the same among the NNMi management servers.
- Edit the `lwssofmconf.xml` file and disable SSO.

If you choose to not complete one of these actions, each time you browse to a different NNMi management server, you will be automatically signed out of the previous NNMi management server. See the *lwssofmconf.xml* reference page, or the UNIX manpage for more information.

If the domain name of the NNMi management server is short, as in `mycompany`, without any dot, the NNMi console will immediately sign you out. The restrictions for SSO browser cookies require a domain name to contain at least one dot, such as `mycompany.com`. To remedy this, complete the following steps:

- 1 Open the following file in a text editor:
 - **Windows:** `;%$NNM_SHARED_CONF%/lwssofmconf.xml`
 - **UNIX:** `$(NNM_SHARED_CONF)/lwssofmconf.xml`

- 2 For this example, search for the following string:

```
<domain>mycompany</domain>
```

and replace it with the following string:

```
domain>myhost.mycompany<domain>
```

- 3 Restart `ovjboss`:
 - a `ovstop ovjboss`
 - b `ovstart ovjboss`

SSO Access for NNMi and the iSPIs

To use SSO, access NNMi as follows:

- Use the correct URL in the following form:
`http://<fully_qualified_domain_name>:<port_number>/nnm/`
<fully_qualified_domain_name> represents the official fully-qualified domain name (FQDN) of the NNMi management server.
<port_number> is the port number assigned during NNMi installation.
- Sign in to NNMi using a valid account.

For SSO to work, URL access to NNMi and the NNM iSPIs must share a common network domain name. Additionally, the URL must not include an IP address. If you do not have a FQDN for the NNMi management server, you can substitute the IP address of the NNMi management server. However, doing so disables single sign-on for NNM iSPIs, and you must sign in again the next time you access any NNM iSPI.

To determine the official FQDN of the NNMi management server, use one of the following methods:

- Use the `nnmofficialfqdn.ovpl` command to display the value of the official FQDN set during installation. See the *nnmofficialfqdn.ovpl* reference page, or the UNIX manpage, for more information.
- In the NNMi console, click **Help > System Information**. On the **Server** tab, look for the official FQDN statement.

If you need to change the official FQDN set during installation, use the `nnmsetofficialfqdn.ovpl` command. See the *nnmsetofficialfqdn.ovpl* reference page, or the UNIX manpage, for more information.



After installation, the system account is still valid. Use the system account only for command-line security and for recovery purposes.

SSO to NNM iSPIs require that users access the NNMi console through a URL that contains the official FQDN. You can configure NNMi to redirect NNMi URLs to the official FQDN when the NNMi console is accessed through a non-official domain name, such as an IP address or a shortened version of the domain name. Before configuring NNMi to redirect URLs, an appropriate official FQDN must be configured. For information, see the NNMi help.

After you enable NNMi to redirect URLs, note the following:

- You can sign on to the NNMi console using any hostname that is valid for the NNMi management server you want to access. For example, if you request `http://localhost/nnm`, NNMi redirects you to a URL such as `http://host.mydomain.com/nnm`.
- If you cannot access the NNMi console, use the following to directly access the NNMi console: `http://server:port/nnm/launch?cmd=showMain`.

Configuring Single Sign-On for the HP NNMi–HP BAC My BSM Integration

Single sign-on is available for all HP enterprise applications that use identical initialization string values and also share a common network domain name.

If the NNMi and BAC user names are exactly the same for a particular individual, that person can log on to the My BSM portal and view NNMi portlets without also signing in to NNMi. This single sign-on feature maps user names, but not passwords, between the two products. The passwords for signing in to My BSM and NNMi can be different. Single sign-on does not map user roles, so the user can have different privileges in each application. For example, a user might have normal privileges in BAC and administrator privileges in NNMi.

To configure single sign-on access from BAC to NNMi, make sure that both applications use the same initialization string. You can copy the string from either application to the other. Consider all applications that interact when choosing which initialization string value to use. If necessary, also update the initialization string configuration for other applications or NNM iSPIs.

BAC initialization string

Locate the BAC initialization string as follows:

- 1 Access the JMX console for BAC at:
`http://<BAC_hostname>:<BAC_JMX_port>/jmx-console/`
- 2 Select **service=LW-SSO Configuration** (under Topaz).
The initialization string is the value of the **InitString** parameter.
- 3 If you change the value of the **InitString** parameter, click **Apply Changes**.

NNMi initialization string

Locate the NNMi initialization string as follows:

- 1 Open the following file in a text editor:
 - *Windows:* `;%$NNM_SHARED_CONF%/lwssofmconf.xml`
 - *UNIX:* `/$NNM_SHARED_CONF/lwssofmconf.xml`
- 2 Search for the string `initString`.

The initialization string is the value of the `initString` parameter without the quotation marks.

For example, if the `lwssofmconf.xml` file contains the following text:

```
initString="E091F3BA8AE47032B3B35F1D40F704B4"
```

the initialization string is:

```
E091F3BA8AE47032B3B35F1D40F704B4
```

- 3 If you change the value of the `initString` parameter, restart `ovjboss`:
 - a **`ovstop ovjboss`**

b ovstart ovjboss

Disabling SSO

If you have a need to disable SSO, complete the following steps:

1 Edit the following file:

- *Windows*: %NNM_SHARED_CONF%\lwssofmconf.xml
- *UNIX*: \$NNM_SHARED_CONF/lwssofmconf.xml

2 Look for look for a section in the file that resembles the following:

```
<enableLWSSO enableLWSSOFramework="true"
enableCookieCreation="true" enableAutoCookieCreation="true"
cookieCreationType="LWSSO" enableSAML2Support="false"/>
```

Change this as follows:

```
<enableLWSSO enableLWSSOFramework="false"
enableCookieCreation="true" enableAutoCookieCreation="true"
cookieCreationType="LWSSO" enableSAML2Support="false"/>
```

3 Run the following commands to restart ovjboss:

- a ovstop ovjboss**
- b ovstart ovjboss**

SSO Security Notes

1 Confidential `initString` parameter in SSO security.

SSO uses *Symmetric Encryption* to validate and create an SSO token. The `initString` parameter within the configuration is used for initialization of the secret key. An application creates a token, and each application that uses the same `initString` parameter validates the token.

The following information is very important:

- It is not possible to use SSO without setting the `initString` parameter.
- The `initString` parameter is confidential information and should be treated as such in terms of publishing, transporting, and persistency.
- Applications that integrate with each other can share the `initString` using SSO.
- The minimum length of the `initString` is 12 characters.

2 Disable SSO unless it is specifically required.

3 The application that uses the weakest authentication framework, and issues an SSO token that is trusted by other integrated applications, determines the level of authentication security for all the applications.

HP recommends that only applications using strong and secure authentication frameworks issue an SSO token.

4 Symmetric encryption implication:

SSO uses symmetric cryptography for issuing and validating SSO tokens. Therefore, any application using SSO can issue a token to be trusted by all other applications sharing the same `initString`.

This potential risk is relevant when an application sharing the `initString` either resides or is accessible in an untrusted location.

5 User mappings (synchronization):

SSO does not ensure user mapping between the integrated applications. Therefore, the integrated application must monitor user mapping. HP recommends you share the same user registry (as LDAP/AD) among all integrated applications.

Failure to map users may cause security breaches and negative application behavior. For example, the same user name may be assigned to different real users in the various applications.

There could be situations when a user logs onto an application (application A) and then accesses a second application (application B) that uses container or application authentication. The failure to map the user will force the user to manually log on to application B and enter a username. If the user enters a different user name than the one used to log on to application A, the following unexpected behavior can arise: If the user subsequently accesses a third application, application C, from application A or application B, then the user will access it using the user names that were used to log onto application A or application B respectively.

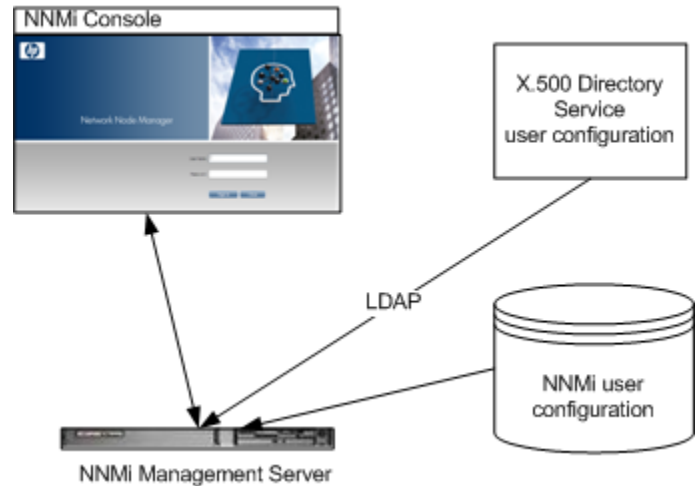
6 Identity Manager is used for an authentication:

All unprotected resources in the Identity Manager must be configured as nonsecure URL settings in the SSO configuration.

7 SSO demo mode:

- Use the SSO demo mode for demonstrative purposes only.
- Only use the demo mode in unsecured networks.
- Do not use the demo mode in production. Any combination of the demo mode with the production mode should not be used.

Integrating NNMi with a Directory Service through LDAP



This chapter contains information about integrating NNMi with a directory service for consolidating the storage of user names, passwords, and, optionally, NNMi role assignments. It contains the following topics:

- [NNMi User Access Information and Configuration Options](#) on page 119
- [Configuring NNMi to Access a Directory Service](#) on page 123
- [Configuring an SSL Connection to the Directory Service](#) on page 110
- [Directory Service Queries](#) on page 128
- [Directory Service Configuration for Storing NNMi Roles](#) on page 139
- [Troubleshooting the Directory Service Integration](#) on page 139
- [ldap.properties Configuration File Reference](#) on page 140

NNMi User Access Information and Configuration Options

Together, the following items define an NNMi user:

- The **user name** uniquely identifies the NNMi user. The user name provides access to NNMi and receives incident assignments.
- The **password** is associated with the user name to control access to the NNMi console or NNMi command.
- The **NNMi role** assignment controls the information available and the type of actions that a user can take in the NNMi console. The role assignment also controls the availability of NNMi commands to the user.

NNMi provides several options for where the NNMi user access information is stored, as described in the following topics. [Table 6](#) indicates the databases that store the NNMi user access information for each configuration option.

Table 6 Options for Storing User Information

Option	User Name	Password	Role Mapping
1	NNMi	NNMi	NNMi
2	Both	Directory Service	NNMi
3	Directory Service	Directory Service	Directory Service

When NNMi is integrated with a directory service for some or all of the user access information, the sign-in and role definition statement on the **Server** tab of the **System Information** window indicates the type of information that was obtained through LDAP queries.

Single sign-on between NNMi and other applications is not dependent on how the NNMi user access information is configured or where this information is stored.

Option 1: All NNMi User Information in the NNMi Database

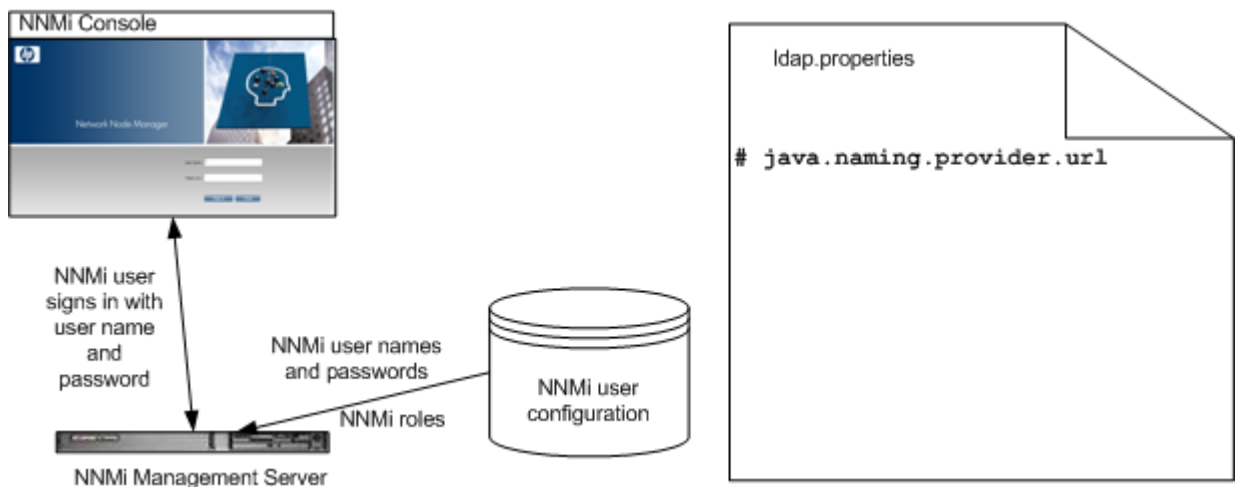
NNMi accesses the NNMi database for all user access information, which the NNMi administrator defines and maintains in the NNMi console. The user access information is local to NNMi. NNMi does not access a directory service, and NNMi ignores the `ldap.properties` file (as indicated by the commented line in [Figure 2](#)).

[Figure 2](#) shows the information flow for this option, which is appropriate in the following situations:

- The number of NNMi users is small.
- No directory service is available.

For information about setting up all user information in the NNMi database, see *Control Access with NNMi Accounts* in the NNMi help. You do not need to read this chapter.

Figure 2 NNMi User Sign-in Information Flow for Option 1



Option 2: Some NNMi User Information in the NNMi Database and Some User Information in the Directory Service

NNMi accesses a directory service for the user name and password, which are defined externally to NNMi and are also available to other applications. The mapping of users to NNMi roles is maintained in the NNMi console. The configuration and maintenance of NNMi user access information is a joint effort as described here:

- The directory service administrator maintains the user names and password in the directory service.
- The NNMi administrator enters the user names (as defined in the directory service) and the NNMi role mappings in the NNMi console.
- The NNMi administrator configures the NNMi `ldap.properties` file to describe the directory service database schema for user names to NNMi. (In [Figure 3](#), the commented line indicates that NNMi does not pull NNMi role information from the directory service.)

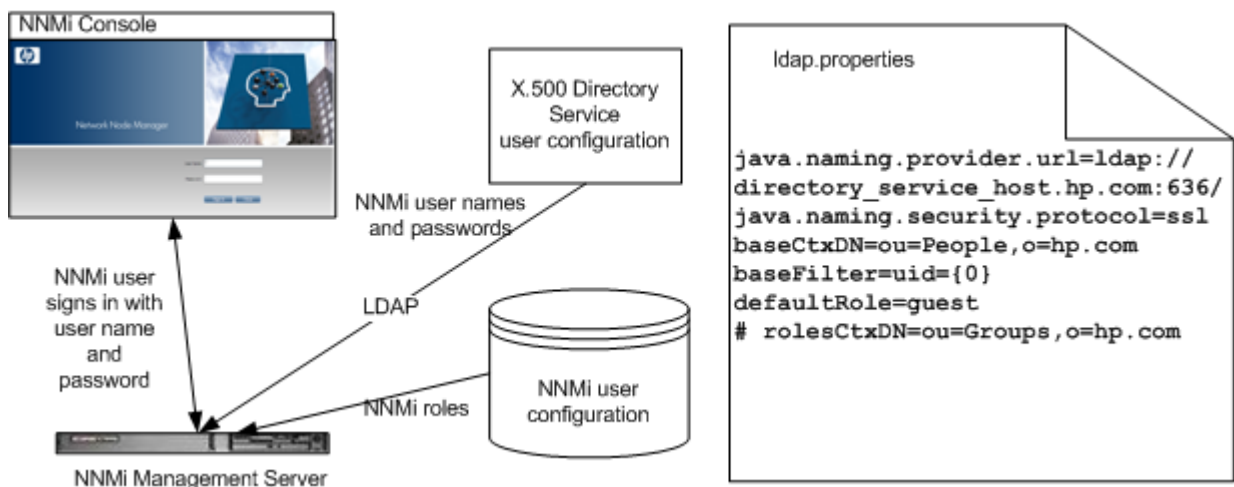
Because user names must be entered in two places, user name maintenance must be performed in both places.

[Figure 3](#) shows the information flow for this option, which is appropriate in the following situations:

- The number of NNMi users is small, and a directory service is available.
- The NNMi administrator wants to control the user roles instead of requiring a directory service change for each role change.
- The directory service group definitions are not easily expandable.

For information about integrating with a directory service for the user name and password, see the rest of this chapter and *Control Access Using Both Directory Service and NNMi* in the NNMi help.

Figure 3 NNMi User Sign-in Information Flow for Option 2



Option 3: All NNMi User Information in the Directory Service

NNMi accesses a directory service for all user access information, which is defined externally to NNMi and is available to other applications. Membership in one or more directory service groups determines the NNMi role for the user. (A user with membership in multiple groups that define NNMi roles receives the most powerful of the roles.)

The directory service administrator maintains the user and group information in the directory service. The directory service administrator configures groups in the directory service, one group for each NNMi role.

The NNMi administrator configures the NNMi `ldap.properties` file to describe the directory service database schema for user names, user groups, and roles to NNMi.

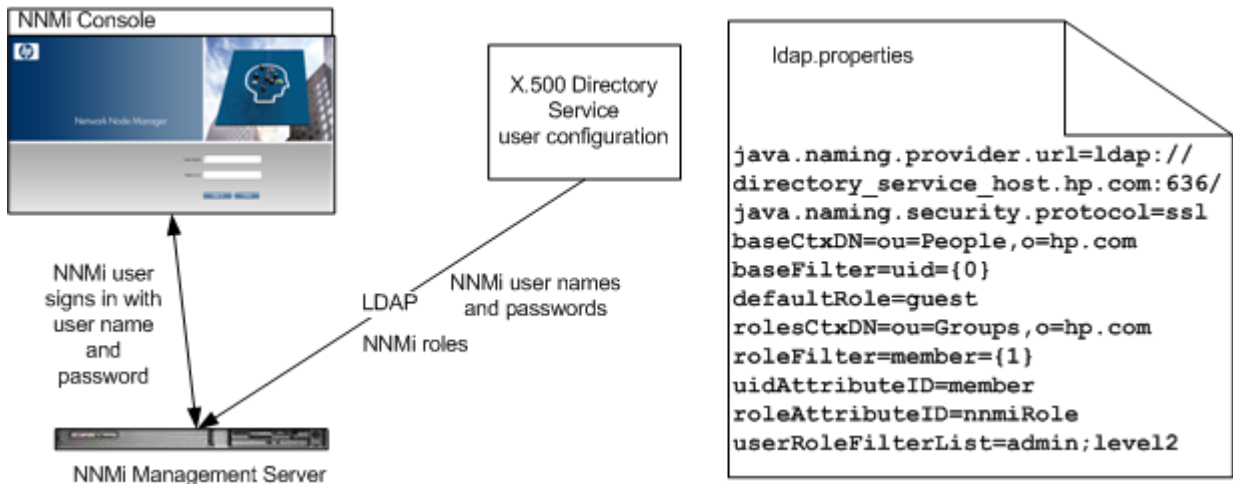
Figure 4 shows the information flow for this option, which is appropriate for environments where the directory service can be modified to include user groups that align with the people who need access to NNMi. Regardless of current configuration, this option requires modifications to the directory service group definitions.

Because this option is an expansion of the option 2 scenario, HP recommends the following configuration process:

- 1 Configure and verify NNMi user name and password retrieval from the directory service.
- 2 Configure NNMi role retrieval from the directory service.

For information about integrating with a directory service for all user information, see the rest of this chapter and *Control Access with a Directory Service* in the NNMi help.

Figure 4 NNMi User Sign-in Information Flow for Option 3



Configuring NNMi to Access a Directory Service

Directory service access is configured in the following file:

- *Windows*: %NNM_SHARED_CONF%\ldap.properties
- *UNIX*: \$NNM_SHARED_CONF/ldap.properties

For information about this file, see [ldap.properties Configuration File Reference](#) on page 140. Also see [Examples](#) on page 144.

For information about the general structure of a directory service, see [Directory Service Queries](#) on page 128.

For configuration option 2, complete the following tasks:

- [Task 1: Back up the Current NNMi User Information](#)
- [Task 2: Optional. Configure Secure Communications to the Directory Service](#)
- [Task 3: Configure NNMi User Access from the Directory Service](#)
- [Task 4: Test the User Name and Password Configuration](#)
- [Task 8: Clean up to Prevent Unexpected Access to NNMi](#)

For configuration option 3, complete the following tasks:

- [Task 1: Back up the Current NNMi User Information](#)
- [Task 2: Optional. Configure Secure Communications to the Directory Service](#)
- [Task 3: Configure NNMi User Access from the Directory Service](#)
- [Task 4: Test the User Name and Password Configuration](#)
- [Task 5: Configure NNMi Role Retrieval from the Directory Service](#)



If you plan to store NNMi roles in the directory service, the directory service must be configured with the NNMi roles. For more information, see [Directory Service Configuration for Storing NNMi Roles](#) on page 139.

- [Task 6: Test the NNMi Role Configuration](#)
- [Task 7: Configure NNMi Roles for Incident Assignment](#)
- [Task 8: Clean up to Prevent Unexpected Access to NNMi](#)

Task 1: [Back up the Current NNMi User Information](#)

Back up the user information in the NNMi database:

- *Windows*: `nmmconfigexport.ovpl -c account -u <user> \ -p <password> -f NNMi_database_accounts.xml`
- *UNIX*: `nmmconfigexport.ovpl -c account -u <user> \ -p <password> -f NNMi_database_accounts.xml`

Task 2: [Optional. Configure Secure Communications to the Directory Service](#)

If the directory service requires the use of secure sockets layer (SSL), import your company's certificate into the NNMi trust store as described in [Configuring an SSL Connection to the Directory Service](#) on page 110.

Task 3: Configure NNMi User Access from the Directory Service

Complete this task for configuration options 2 and 3. Follow the appropriate procedure for your directory service. This task includes the following sections:

- [Simple Approach for Microsoft Active Directory](#)
- [Simple Approach for Other Directory Services](#)

Simple Approach for Microsoft Active Directory

- 1 Back up the `ldap.properties` file that was shipped with NNMi, and then open the file in any text editor.
- 2 Overwrite the file contents with the following text:

```
java.naming.provider.url=ldap://<myldapserver>:389/

bindDN=<mydomain>\\<myusername>
bindCredential=<mypassword>

baseCtxDN=CN=Users,DC=<myhostname>,DC=<mycompanyname>,DC=<mysuffix>
baseFilter=CN={0}

defaultRole=guest

#rolesCtxDN=CN=Users,DC=<myhostname>,DC=<mycompanyname>,DC=<mysuffix>
roleFilter=member={1}
uidAttributeID=member
roleAttributeIsDN=true
roleNameAttributeID=info
roleAttributeID=memberOf
userRoleFilterList=admin;level2;level1
```

- 3 In the following line:

```
java.naming.provider.url=ldap://<myldapserver>:389/
```

Replace `<myldapserver>` with the fully-qualified hostname of the Active Directory server (for example: `myserver.example.com`).



To specify multiple directory service URLs, separate each URL with a single space character.

- 4 In the following lines:

```
bindDN=<mydomain>\\<myusername>
bindCredential=<mypassword>
```

Make the following substitutions:

- Replace `<mydomain>` with the name of Active Directory domain.
- Replace `<myusername>` and `<mypassword>` with a user name and password for accessing the Active Directory server. Because the password is stored in plain text, specify a user name with read-only access to the directory service.

- 5 In the following line:

```
baseCtxDN=CN=Users,DC=<myhostname>,DC=<mycompanyname>,
DC=<mysuffix>
```

Replace `<myhostname>`, `<mycompanyname>`, and `<mysuffix>` with the components of the fully-qualified hostname of the Active Directory server (for example, for the hostname `myserver.example.com`, specify: `DC=myserver,DC=example,DC=com`).

Simple Approach for Other Directory Services

- 1 Back up the `ldap.properties` file that was shipped with NNMi, and then open the file in any text editor.
- 2 In the following line:

```
#java.naming.provider.url=ldap://<myldapserver>:389/
```

Do the following:

- Uncomment the line (by deleting the # character).
- Replace `<myldapserver>` with the fully-qualified hostname of the directory server (for example: `myserver.example.com`).



To specify multiple directory service URLs, separate each URL with a single space character.

- 3 In the following line:

```
baseCtxDN=ou=People,o=myco.com
```

Replace `ou=People,o=myco.com` with the portion of the directory service domain that stores user records.

- 4 In the following line:

```
baseFilter=uid={0}
```

Replace `uid` with the user name attribute from the directory service domain.

Task 4: Test the User Name and Password Configuration

- 1 In the `ldap.properties` file, set `defaultRole=guest` for testing purposes. (You can change this value at any time.)
- 2 Save the `ldap.properties` file.
- 3 Force NNMi to re-read the `ldap.properties` file by running the following command:

```
nnmldap.ovpl -reload
```

- 4 Sign in to the NNMi console with a user name and password that are defined in the directory service.



Run this test with a user name that is not already defined in the NNMi database.

- 5 Verify the user name and role (guest) in the title bar of the NNMi console.

If user sign in does not work correctly, check the configuration, and then repeat [step 1](#) through [step 5](#) until you see the expected result.

- a Verify that you completed [Task 3](#) on page 124 correctly.
- b Follow the detailed configuration process in [User Identification](#) on page 133.



After each test, sign out of the NNMi console to clear the session credentials.

- 6 After you can sign in, choose your strategy:
 - If you plan to store role assignments in the NNMi database (configuration option 2), continue with [Task 8](#) on page 127.
 - If you plan to store role assignments in the directory service (configuration option 3), continue with [Task 5](#).

Task 5: [Configure NNMi Role Retrieval from the Directory Service](#)

Complete this task for configuration option 3. Follow the appropriate procedure for your directory service. This task includes the following sections:

- [Simple Approach for Microsoft Active Directory](#)
- [Simple Approach for Other Directory Services](#)

[Simple Approach for Microsoft Active Directory](#)

- 1 Open the `ldap.properties` file in any text editor.
- 2 In the following line:

```
#rolesCtxDN=CN=Users,DC=<myhostname>,DC=<mycompanyname>,
DC=<mysuffix>
```

Do the following:

- Uncomment the line (by deleting the # character).
- Replace `<myhostname>`, `<mycompanyname>`, and `<mysuffix>` with the components of the fully-qualified hostname of the Active Directory server (for example, for the hostname `myserver.example.com`, specify:
`DC=myserver,DC=example,DC=com`).

[Simple Approach for Other Directory Services](#)

- 1 Open the `ldap.properties` file in any text editor.
- 2 In the following line:

```
#rolesCtxDN=ou=Groups,o=myco.com
```

Do the following:

- Uncomment the line (by deleting the # character).
- Replace `ou=Groups, o=myco.com` with the portion of the directory service domain that stores group records.

- 3 In the following line:

```
roleFilter=member={1}
```

Replace `member` with the name of the group attribute that stores the directory service user ID in the directory service domain.

Task 6: [Test the NNMi Role Configuration](#)

- 1 Save the `ldap.properties` file.
- 2 Force NNMi to re-read the `ldap.properties` file by running the following command:

```
nnmldap.ovpl -reload
```

- 3 Sign in to the NNMi console with a user name and password that are defined in the directory service.



Run this test with a user name that is not already defined in the NNMi database.

- 4 Verify the user name and role (as configured in the directory service) in the title bar of the NNMi console.

If user signin does not work correctly, check the configuration, and then repeat [step 1](#) through [step 4](#) until you see the expected result.

- a Verify that you completed [Task 5](#) on page 126 correctly.
- b Follow the detailed configuration process in [Role Identification](#) on page 135.



After each test, sign out of the NNMi console to clear the session credentials.

Task 7: Configure NNMi Roles for Incident Assignment

- 1 Open the `ldap.properties` file in any text editor.
- 2 Modify the `userRoleFilterList` parameter value to specify the NNMi roles to which NNMi operators can assign incidents.
- 3 Save the `ldap.properties` file.
- 4 Force NNMi to re-read the `ldap.properties` file by running the following command:

```
nnmldap.ovpl -reload
```
- 5 Sign in to the NNMi console with a user name and password that are defined in the directory service.
- 6 In any incident view, select an incident, and then click **Actions > Assign > Assign Incident**. Verify that you can assign the incident to a user in each of the roles specified by the `userRoleFilterList` parameter.
- 7 Repeat [step 1](#) through [step 6](#) until you can assign an incident to each configured role type.

Task 8: Clean up to Prevent Unexpected Access to NNMi

- 1 Optional. Change the value of, or comment out, the `defaultRole` parameter in the `ldap.properties` file.
- 2 Reset the user access information in the NNMi database:
 - To store role assignments in the NNMi database (configuration option 2), do the following in the NNMi console:
 - Remove any pre-existing user access information. (Delete all objects on the **User Accounts** and **User Principals** tabs of the **User Interface Configuration** form.)
 - Create role assignments for the directory service user names. (For each NNMi user, create a new object on the **User Accounts** tab of the **User Interface Configuration** form that associates the user name (as defined in the directory service) with an NNMi role.)
 - Update incident ownership so that each assigned incident is associated with a valid user name.

For instructions, see *Control Access Using Both Directory Service and NNMi* in the NNMi help.

- If you configured role assignment from the directory service (configuration option 3), do the following in the NNMi console:
 - Remove any pre-existing user access information. (Delete all objects on the **User Accounts** and **User Principals** tabs of the **User Interface Configuration** form.)
 - Update incident ownership so that each assigned incident is associated with a valid user name.

For instructions, see *Control Access with a Directory Service* in the NNMi help.

Directory Service Queries

NNMi uses LDAP to communicate with a directory service. NNMi sends a request, and the directory service returns stored information. NNMi cannot alter the information that is stored in the directory service.

This section contains the following topics:

- [Directory Service Access](#)
- [Directory Service Content](#)
- [Information Owned by the Directory Service Administrator](#)
- [User Identification](#)
- [Role Identification](#)

Directory Service Access

LDAP queries to a directory service use the following format:

ldap://<directory_service_host>:<port>/<search_string>

- `ldap` is the protocol indicator. Use this indicator for both standard connections and SSL connections to the directory service.
- `<directory_service_host>` is the fully-qualified name of the computer that hosts the directory service.
- `<port>` is the port that the directory service uses for LDAP communication. The default port for non-SSL connections is 389. The default port for SSL connections is 636.
- `<search_string>` contains the information request. For more information, see [Directory Service Content](#) and RFC 1959, *An LDAP URL Format*, which is available at:
labs.apache.org/webarch/uri/rfc/rfc1959.txt

You can enter an LDAP query as a URL in a web browser to verify that you have the correct access information and the correct structure for the search string.



If the directory service (for example, Active Directory) does not allow anonymous access, the directory service denies LDAP queries from a web browser. In this case, you can use a commercially available LDAP browser to validate your configuration parameters.

Directory Service Content

A directory service stores information such as user names, passwords, and group membership. To access the information in a directory service, you must know the distinguished name that references the storage location of the information. For sign-in applications, the distinguished name is a combination of variable information (such as a user name) and fixed information (such as the storage location of user names). The elements that make up a distinguished name depend on the structure and content of the directory service.

The following examples show possible definitions for a group of users called USERS-NNMi-Admin. This group lists the directory service user IDs that have administrative access to NNMi. The following information pertains to these examples:

- The Active Directory example is for the Windows operating system.
- The other directory services example is for UNIX operating systems.
- The file shown in each example is a portion of a lightweight directory interchange format (LDIF) file. LDIF files provide for sharing directory service information.
- Each example also includes a figure that shows a graphical representation of the directory service domain, which provides an expanded view of the information in the LDIF file excerpt.

Example content
structure for Active
Directory

In this example, the following items are of interest:

- The distinguished name of the user John Doe is:
CN=john.doe@example.com,OU=Users,OU=Accounts,DC=example,DC=com
- The distinguished name of the group USERS-NNMi-Admin is:
CN=USERS-NNMi-Admin,OU=Groups,OU=Accounts,DC=example,DC=com
- The group attribute that stores the directory service user ID is: `member`
- The group attribute that stores the NNMi role is: `info`

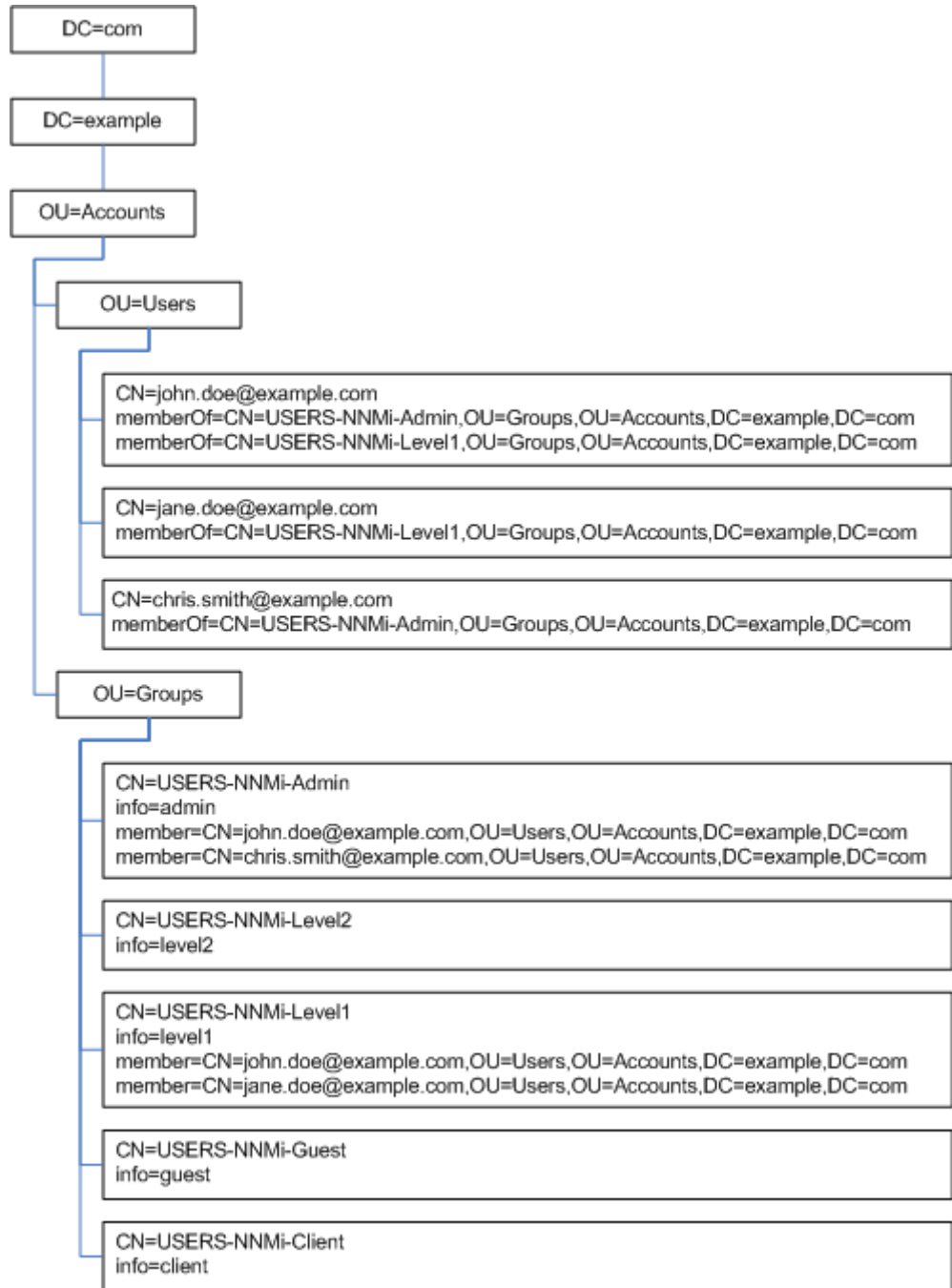
The `info` attribute has been repurposed to store the NNMi role.

Example LDIF file excerpt:

```
groups |USERS-NNMi-Admin
dn: CN=USERS-NNMi-Admin,OU=Groups,OU=Accounts,DC=example,DC=com
info: admin
cn: USERS-NNMi-Admin
description: Group of users for NNMi administration.
member: CN=john.doe@example.com,OU=Users,OU=Accounts,
DC=example,DC=com
member: CN=chris.smith@example.com,OU=Users,OU=Accounts,
DC=example,DC=com
```

Figure 5 on page 130 illustrates this directory service domain.

Figure 5 Example Domain for Active Directory



Example content structure for other directory services

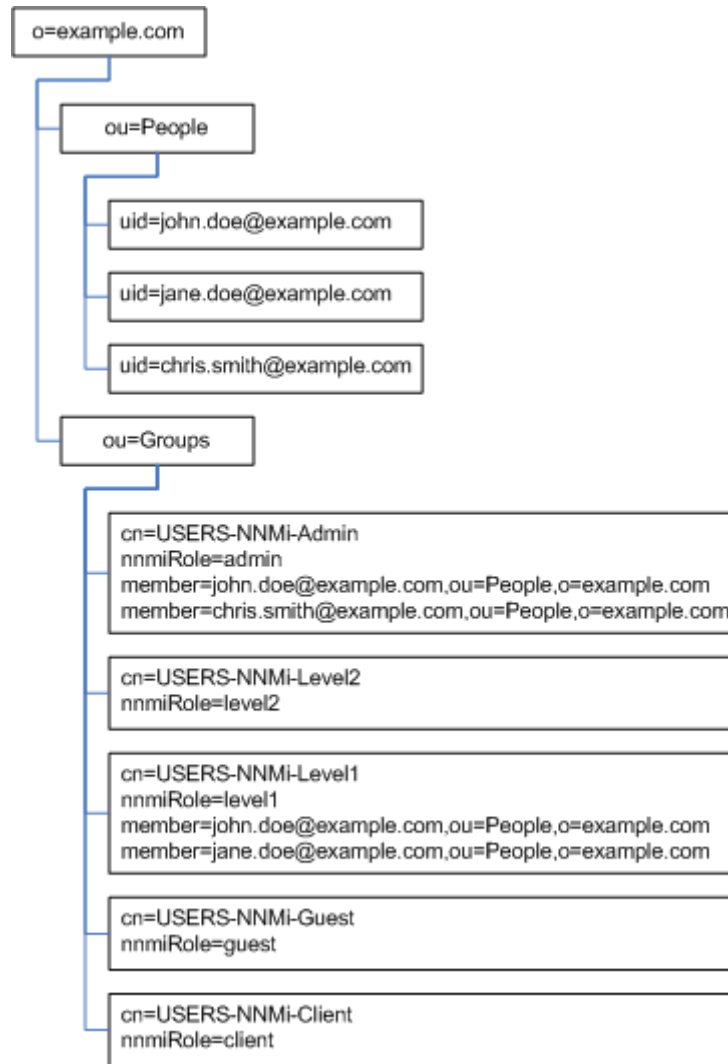
In this example, the following items are of interest:

- The distinguished name of the user John Doe is:
uid=john.doe@example.com,ou=People,o=example.com
- The distinguished name of the group USERS-NNMi-Admin is:
cn=USERS-NNMi-Admin,ou=Groups,o=example.com
- The group attribute that stores the directory service user ID is: member
- The group attribute that stores the NNMi role is: nnmiRole
The nnmiRole attribute was created specifically for this purpose.

Example LDIF file excerpt:

```
groups |USERS-NNMi-Admin
dn: cn=USERS-NNMi-Admin,ou=Groups,o=example.com
nnmiRole: admin
cn: USERS-NNMi-Admin
description: Group of users for NNMi administration.
member: uid=john.doe@example.com,ou=People,o=example.com
member: uid=chris.smith@example.com,ou=People,o=example.com
```

Figure 6 Example Domain for Other Directory Services



Information Owned by the Directory Service Administrator

Table 7 and Table 8 list the information to obtain from the directory service administrator before configuring NNMi for LDAP access to a directory service.

- If you plan to use the directory service for user names and passwords only (configuration option 2), gather the information for Table 7.
- If you plan to use the directory service for all NNMi access information (configuration option 3), gather the information for Table 7 and Table 8.

Table 7 Information for Retrieving User Names and Passwords from a Directory Service

Information	Active Directory Example	Other Directory Services Example
The fully-qualified name of the computer that hosts the directory service	directory_service_host.example.com	
The port that the directory service uses for LDAP communication	<ul style="list-style-type: none"> • 389 for non-SSL connections • 636 for SSL connections 	
Does the directory service require an SSL connection?	If yes, obtain a copy of your company's trust store certificate and see Configuring an SSL Connection to the Directory Service on page 110.	
The distinguished name for one user name that is stored in the directory service (to demonstrate the directory service domain)	CN=john.doe@example.com, OU=Users,OU=Accounts, DC=example,DC=com	uid=john.doe@example.com, ou=People,o=example.com

Table 8 Information for Retrieving NNMi Roles from a Directory Service

Information	Active Directory Example	Other Directory Services Example
The distinguished name for identifying the groups to which a user is assigned	The memberOf user attribute identifies the groups.	<ul style="list-style-type: none"> • ou=Groups,o=example.com • cn=USERS-NNMi-*, ou=Groups,o=example.com
The method of identifying a user within a group	<ul style="list-style-type: none"> • CN=john.doe@example.com, OU=Users,OU=Accounts, DC=example,DC=com • CN=john.doe@example.com 	<ul style="list-style-type: none"> • cn=john.doe@example.com, ou=People,o=example.com • cn=john.doe@example.com
The group attribute that stores the directory service user ID	member	member

Table 8 Information for Retrieving NNMi Roles from a Directory Service (cont'd)

Information	Active Directory Example	Other Directory Services Example
The names of the groups that the directory service administrator has created for mapping users to NNMi roles	<ul style="list-style-type: none"> • CN=USERS-NNMi-Admin, OU=Groups, OU=Accounts, DC=example, DC=com • CN=USERS-NNMi-Level2, OU=Groups, OU=Accounts, DC=example, DC=com • CN=USERS-NNMi-Level1, OU=Groups, OU=Accounts, DC=example, DC=com • CN=USERS-NNMi-Client, OU=Groups, OU=Accounts, DC=example, DC=com • CN=USERS-NNMi-Guest, OU=Groups, OU=Accounts, DC=example, DC=com 	<ul style="list-style-type: none"> • cn=USERS-NNMi-Admin, ou=Groups, o=example.com • cn=USERS-NNMi-Level2, ou=Groups, o=example.com • cn=USERS-NNMi-Level1, ou=Groups, o=example.com • cn=USERS-NNMi-Client, ou=Groups, o=example.com • cn=USERS-NNMi-Guest, ou=Groups, o=example.com
The name of the attribute in each of the new groups that stores the NNMi role	info	nnmiRole

User Identification

User identification applies to configuration options 2 and 3.

The distinguished name for user identification is the fully-qualified method of locating one user in the directory service. NNMi passes the user distinguished name in an LDAP request to the directory service.

In the `ldap.properties` file, the user distinguished name is the concatenation of the `baseFilter` value and the `baseCtxDN` value. If the password returned by the directory service matches the sign-in password that the user entered into the NNMi console, then user `signin` continues.

For configuration option 2, NNMi examines the following information and grants the user the role with the more privileges:

- The value of the `defaultRole` parameter in the `ldap.properties` file
- The role assigned to this user in the NNMi console

For configuration option 3, NNMi determines the user role as described in [Role Identification](#) on page 135.

Active Directory user identification example

If `baseFilter` is set to `CN={0}`, `baseCtxDN` is set to `OU=Users, OU=Accounts, DC=example, DC=com`, and a user signs in to NNMi as `john.doe`, then the string passed to the directory service is:

```
CN=john.doe, OU=Users, OU=Accounts, DC=example, DC=com
```

Other directory services user identification example

If `baseFilter` is set to `uid={0}@example.com`, `baseCtxDN` is set to `ou=People, o=example.com`, and a user signs in to NNMi as `john.doe`, then the string passed to the directory service is:

```
uid=john.doe@example.com, ou=People, o=example.com
```

Configuring NNMi User Access from the Directory Service (Detailed Approach)

If the simple approach described in [Task 3](#) on page 124 did not work correctly, follow these steps:

- 1 Obtain the information listed in [Table 7](#) on page 132 from the directory service administrator.
- 2 Complete the appropriate procedure:
 - *LDAP browser approach for Active Directory and other directory services:* See [Determining How the Directory Service Identifies a User \(LDAP Browser Approach\)](#) on page 135.
 - *Web browser approach for other directory services:* See [Determining How the Directory Service Identifies a User \(Web Browser Approach\)](#) on page 135.
- 3 Open the `ldap.properties` file in any text editor.



For information about the `ldap.properties` file, see [ldap.properties Configuration File Reference](#) on page 140.

- 4 Set the `java.naming.provider.url` parameter to the URL for accessing the directory service through LDAP.
 - *LDAP browser approach:* Obtain this information from the LDAP browser configuration.
 - *Web browser approach:* Include the values of `<directory_service_host>` and `<port>` from [Determining How the Directory Service Identifies a User \(Web Browser Approach\)](#) on page 135.



To specify multiple directory service URLs, separate each URL with a single space character.

- 5 If you configured secure communications to the directory service, uncomment (or add) the following line:

```
java.naming.security.protocol=ssl
```

- 6 (Active Directory) Set the `bindDN` and `bindCredential` parameters as follows:
 - Replace `<mydomain>` with the name of Active Directory domain.
 - Replace `<myusername>` and `<mypassword>` with a user name and password for accessing the Active Directory server. Because the password is stored in plain text, specify a user name with read-only access to the directory service.
- 7 Set the `baseCtxDN` parameter to the elements of the distinguished user name that are the same for multiple users.
- 8 Set the `baseFilter` parameter to correlate user names as they are entered for NNMi sign-in to the way user names are stored in the directory service.

This value is the element of the distinguished user name that changes for each user. Replace the actual user name with the expression `{0}`.
- 9 Test the configuration as described in [Task 4](#) on page 125.

Determining How the Directory Service Identifies a User (LDAP Browser Approach)

In a third-party LDAP browser, do the following:

- 1 Navigate to the portion of the directory service domain that stores group information.
- 2 Identify a group of users, and then examine the format of the distinguished names for the users associated with that group.

Determining How the Directory Service Identifies a User (Web Browser Approach)

- 1 In a supported web browser, enter the following URL:
ldap://<directory_service_host>:<port>/<user_search_string>
 - <directory_service_host> is the fully-qualified name of the computer that hosts the directory service.
 - <port> is the port that the directory service uses for LDAP communication.
 - <user_search_string> is the distinguished name for one user name that is stored in the directory service.
- 2 Evaluate the results of the directory service access test.
 - If the request times out or you see a message that the directory service could not be reached, verify the values of <directory_service_host> and <port>, and then repeat [step 1](#).
 - If you see a message that the directory service does not contain the requested entry, verify the value of <user_search_string>, and then repeat [step 1](#).
 - If you see the appropriate user record, the access information is correct. The value of <user_search_string> is the distinguished user name.

Role Identification

Role identification applies to configuration option 3.

NNMi determines a user's NNMi role by examining the user's group membership within the directory service. In the directory service groups and the NNMi configuration files, short text strings identify the NNMi roles. [Table 9](#) maps the text strings to the role names that appear in the NNMi console.

Table 9 NNMi Role Name Mappings

Role Name in the NNMi Console	Text String in Directory Service Groups and NNMi Configuration Files
Administrator	admin
Operator Level 2	level2
Operator Level 1	level1
Guest	guest
Web Service Client	client

Active Directory Role Identification

A user definition in the directory service domain includes the directory service groups to which the user belongs. Each group name is indicated by a separate entry of a certain user attribute, commonly called `memberOf`. In the `ldap.properties` file, the `roleAttributeID` parameter specifies the name of the user attribute that stores the user's group membership.

For each listed group, NNMi determines whether that group definition includes an NNMi role. In the `ldap.properties` file, the `roleNameAttributeID` parameter specifies the name of the group attribute that stores the text string for an NNMi role (as defined in [Table 9](#)). NNMi retrieves the value of that attribute to determine what role to assign to the NNMi user. (When a user is assigned multiple roles, NNMi uses the role with more privileges.)

Other Directory Services Role Identification

A group definition in the directory service domain includes the directory service users who belong to that group.

In the `ldap.properties` file, the `roleFilter` value specifies how group members are identified, and the `rolesCtxDN` value specifies where in the directory service domain the groups are defined. For example, if `roleFilter` is set to `member={1}`, `rolesCtxDN` is set to `ou=Groups,o=example.com`, and the authenticated user distinguished name is `john.doe@example.com`, then NNMi queries the directory services for all groups whose fully-qualified distinguished names end in `ou=Groups,o=example.com`.

NNMi parses the list of groups returned by the directory service to determine which groups include the specification:

```
member=uid=john.doe@example.com,ou=People,o=example.com
```

Finally, NNMi parses the list of groups with the appropriate `member` specification to determine if any group has the attribute that corresponds to an NNMi role.

In the `ldap.properties` file, the value of `roleAttributeID` is the name of the group attribute that stores the text string for an NNMi role (as defined in [Table 9](#)). NNMi retrieves the value of that attribute to determine what role to assign to the NNMi user. (When a user is assigned multiple roles, NNMi uses the role with more privileges.)

In the `ldap.properties` file, the value of `uidAttributeID` is the name of the group attribute that stores the user name. NNMi retrieves the value of that attribute to cross-check the configuration.

Configuring NNMi Role Retrieval from the Directory Service (Detailed Approach)

If the simple approach described in [Task 5](#) on page 126 did not work correctly, follow these steps:

- 1 Obtain the information listed in [Table 8](#) on page 132 from the directory service administrator.
- 2 Complete the appropriate procedure:
 - *LDAP browser approach for Active Directory*: See [Determining How the Directory Service Identifies a Group and Group Membership \(LDAP Browser Approach for Active Directory\)](#) on page 138.
 - *LDAP browser approach for other directory services*: See [Determining How the Directory Service Identifies a Group and Group Membership \(LDAP Browser Approach for Other Directory Services\)](#) on page 138.
 - *Web browser approach for other directory services*: See [Determining How the Directory Service Identifies a Group \(Web Browser Approach\)](#) on page 138.
- 3 Open the `ldap.properties` file in any text editor.



For information about the `ldap.properties` file, see [ldap.properties Configuration File Reference](#) on page 140.

- 4 Set the `rolesCtxDN` parameter to the elements of the distinguished group name that are the same for multiple groups.
- 5 Set the `roleFilter` parameter to correlate user names to the way user names are stored for groups in the directory service. Replace the actual user name with one of the following expressions:
 - Use `{0}` to denote the user name entered for signin (for example, `john.doe`).
 - Use `{1}` to denote the distinguished name of the authenticated user as returned by the directory service (for example, `uid=john.doe@example.com,ou=People,o=example.com`).
- 6 Set the `uidAttributeID` parameter to the name of the group attribute that stores the user ID.
- 7 Set the `roleAttributeID` parameter as follows:
 - For Active Directory, specify the name of the *user* attribute that indicates the groups to which the user belongs.
 - For other directory services, specify the name of the *group* attribute that stores the NNMi role text string (one of the values listed in the second column of [Table 9](#) on page 135).
- 8 Set the `roleAttributeIsDN` parameter as follows:
 - For Active Directory, set this parameter to `true`.
 - For other directory services, set this parameter to `false`.
- 9 (Active Directory) Set the `roleNameAttributeID` parameter to the name of the *group* attribute that stores the NNMi role text string (one of the values listed in the second column of [Table 9](#) on page 135).
- 10 Test the configuration as described in [Task 6](#) on page 126.

Determining How the Directory Service Identifies a Group and Group Membership (LDAP Browser Approach for Active Directory)

In a third-party LDAP browser, do the following:

- 1 Navigate to the portion of the directory service domain that stores user information.
- 2 Identify a user who requires access to NNMi, and then examine the format of the distinguished names for the groups associated with that user.
- 3 Navigate to the portion of the directory service domain that stores group information.
- 4 Identify the groups that correspond to NNMi roles, and then examine the format of the names for the users associated with a group.

Determining How the Directory Service Identifies a Group and Group Membership (LDAP Browser Approach for Other Directory Services)

In a third-party LDAP browser, do the following:

- 1 Navigate to the portion of the directory service domain that stores group information.
- 2 Identify the groups that correspond to NNMi roles, and then examine the format of the distinguished names for those groups.
- 3 Also examine the format of the names for the users associated with a group.

Determining How the Directory Service Identifies a Group (Web Browser Approach)

- 1 In a supported web browser, enter the following URL:
`ldap://<directory_service_host>:<port>/<group_search_string>`
 - `<directory_service_host>` is the fully-qualified name of the computer that hosts the directory service.
 - `<port>` is the port that the directory service uses for LDAP communication.
 - `<group_search_string>` is the distinguished name for a group name that is stored in the directory service, for example:
`cn=USERS-NNMi-Admin,ou=Groups,o=example.com`
- 2 Evaluate the results of the directory service access test.
 - If you see a message that the directory service does not contain the requested entry, verify the value of `<group_search_string>`, and then repeat [step 1](#).
 - If you see the appropriate list of groups, the access information is correct.
- 3 Examine the group properties to determine the format of the names for the users associated with that group.

Directory Service Configuration for Storing NNMi Roles

If you plan to store NNMi roles in the directory service (configuration option 3), the directory service must be configured with NNMi role information. Ideally, the directory service already contains appropriate user groups. If this is not the case, the directory service administrator can create new user groups specifically for NNMi role assignment. Either way, the directory service administrator must maintain a group attribute for the NNMi role. This new attribute corresponds to the `roleNameAttributeID` parameter (for Active Directory) or to the `roleAttributeID` parameter (for other directory services) in the `ldap.properties` file. [Table 9](#) on page 135 lists the possible role attribute values in the directory service.

Because directory service configuration and maintenance procedures depend on the specific directory service software and your company's policies, those procedures are not documented here.

Troubleshooting the Directory Service Integration

- 1 Verify the NNMi LDAP configuration by running the following command:

```
nnmlldap.ovpl -info
```

If the reported configuration is not as expected, verify the settings in the `ldap.properties` file.

- 2 Force NNMi to re-read the `ldap.properties` file by running the following command:

```
nnmlldap.ovpl -reload
```

- 3 Make sure that the directory service contains the expected records. Use a web browser or a third-party LDAP browser to examine the directory service information.

Information about the format of a query to a directory service can be found in RFC 1959, *An LDAP URL Format*, which is available at:

```
http://labs.apache.org/webarch/uri/rfc/rfc1959.txt
```

ldap.properties Configuration File Reference

The `ldap.properties` file contains the settings for communicating with and building LDAP queries to the directory service. This file is located as follows:

- **Windows:** `%NNM_SHARED_CONF%\ldap.properties`
- **UNIX:** `$NNM_SHARED_CONF/ldap.properties`

In the `ldap.properties` file, the following conventions apply:

- To comment out a line, begin that line with a number sign character (`#`).
- The following rules apply to special characters:
 - To specify a backslash character (`\`), comma (`,`), semi-colon (`;`), plus sign (`+`), less than sign (`<`), or greater than sign (`>`), escape the character with a backslash character. For example: `\\` or `\+`
 - To include a space character () as the *first* or *last* character in a string, escape the space character with a backslash character (`\`).
 - To include a number sign character (`#`) as the *first* character in a string, escape the number sign character with a backslash character (`\`).

Characters not mentioned here do not need to be escaped or quoted.

▶ After editing the `ldap.properties` file, force NNMi to re-read the LDAP configuration by running the following command:

```
nnmlldap.ovpl -reload
```

Table 10 describes the parameters in the `ldap.properties` file.

▶ The initial `ldap.properties` file might not include all parameters that are listed in Table 10. Add the parameters that you need.

Table 10 Parameters in the ldap.properties File

Parameter	Description
<code>java.naming.provider.url</code>	<p>The URL for accessing the directory service.</p> <p>The format is the protocol (<code>ldap</code>), followed by the fully-qualified host name of the directory server, optionally followed by the port number. For example:</p> <pre>java.naming.provider.url=ldap://ldap.example.com:389/</pre> <p>If the port number is omitted the following defaults apply:</p> <ul style="list-style-type: none">• For non-SSL connections, the default port is 389.• For SSL connections, the default port is 636. <p>If you specify multiple directory service URLs, NNMi uses the first directory service when possible. If that directory service is not accessible, NNMi queries the next directory service in the list, and so forth. Separate each URL with a single space character. For example:<pre>java.naming.provider.url=ldap://ldap1.example.com/ ldap:// ldap2.example.com/</pre><p>Configuring this parameter enables LDAP communication between NNMi and the directory service. To disable LDAP communication, comment out this parameter, and then save the file. NNMi ignores the configuration in the <code>ldap.properties</code> file.</p></p>

Table 10 Parameters in the ldap.properties File (cont'd)

Parameter	Description
java.naming.security.protocol	<p>The connection protocol specification.</p> <ul style="list-style-type: none"> If the directory service is configured to use LDAP over SSL, set this parameter to <code>ssl</code>. For example: <code>java.naming.security.protocol=ssl</code> If the directory service does not require SSL, leave this parameter commented out. <p>For more information, see Configuring an SSL Connection to the Directory Service on page 110.</p>
bindDN	<p>For a directory service (such as Active Directory) that does not allow anonymous access, identifies the user name for accessing the directory service. Because the password for this user name is stored in plain text in the <code>ldap.properties</code> file, select a user name with read-only access to the directory service.</p> <p>For example: <code>bindDN=region1\john.doe@example.com</code></p>
bindCredential	<p>When <code>bindDN</code> is set, specifies the password for the user name that <code>bindDN</code> identifies. For example: <code>bindCredential=PasswordForJohnDoe</code></p>
baseCtxDN	<p>Identifies the portion of the directory service domain that stores user records.</p> <p>The format is a comma-separated list of directory service attribute names and values. For example:</p> <ul style="list-style-type: none"> <code>baseCtxDN=CN=Users,DC=ldapserver,DC=example,DC=com</code> <code>baseCtxDN=ou=People,o=example.com</code> <p>For more information, see User Identification on page 133.</p>
baseFilter	<p>Identifies the user who is signing in to NNMi.</p> <p>The format is the name of the directory service user name attribute and a string that relates the entered user sign-in name to the format of names in the directory service. The user name string contains the expression <code>{0}</code> (to denote the user name entered for signin) and any other characters that are needed to match the directory service formatting of user names.</p> <ul style="list-style-type: none"> If the user name entered for NNMi signin is the same as the user name stored in the directory service, the value is the replacement expression. For example: <code>baseFilter=CN={0}</code> <code>baseFilter=uid={0}</code> If the user name entered for NNMi signin is as subset of the user name stored in the directory service, include the additional characters in the value. For example: <code>baseFilter=CN={0}@example.com</code> <code>baseFilter=uid={0}@example.com</code> <p>For more information, see User Identification on page 133.</p>

Table 10 Parameters in the ldap.properties File (cont'd)

Parameter	Description
defaultRole	<p>Optional. Specifies a default role that applies to any directory service user who signs in to NNMi through LDAP. The value of this parameter applies regardless of where role mappings are stored (in the NNMi database or in the directory service). If a user is configured for an NNMi role, NNMi uses the role with more privileges.</p> <p>Valid values are as follows: <code>admin</code>, <code>level2</code>, <code>level1</code>, or <code>guest</code>.</p> <p>For example:</p> <pre>defaultRole=guest</pre> <p>If commented out or omitted, NNMi does not use a default role.</p>
rolesCtxDN	<p>Identifies the portion of the directory service domain that stores group records.</p> <p>The format is a comma-separated list of directory service attribute names and values. For example:</p> <ul style="list-style-type: none">• <code>rolesCtxDN=CN=Users,DC=ldapserver,DC=example,DC=com</code>• <code>rolesCtxDN=ou=Groups,o=example.com</code> <p>In other directory services (not Active Directory), for a faster search, you can identify one or more directory service groups that contain NNMi roles. If the group names form a pattern, you can specify a wildcard. For example, if the directory service includes groups named <code>USERS-NNMi-administrators</code>, <code>USERS-NNMi-level1Operators</code>, and so forth, you could use a search context similar to:</p> <pre>rolesCtxDN=cn=USERS-NNMi-*,ou=Groups,o=example.com</pre> <p>Configuring this parameter enables directory service queries for NNMi role assignments through LDAP. To disable directory service queries for NNMi role assignments through LDAP, comment out this parameter, and then save the file. NNMi ignores the remaining role-related values in the <code>ldap.properties</code> file.</p> <p>For more information, see Role Identification on page 135.</p>

Table 10 Parameters in the ldap.properties File (cont'd)

Parameter	Description
roleFilter	<p>Identifies the user who is signing in to NNMi, formatted as the directory service stores group member names.</p> <p>The format is the name of the directory service group attribute for user ID and a string that relates the entered user sign-in name to the format of user IDs in the directory service. The user name string contains one of the following expressions and any other characters that are needed to match the directory service formatting of group member names.</p> <ul style="list-style-type: none"> • The expression <code>{0}</code> denotes the user name entered for signin (for example, <code>john.doe</code>). An example role filter that matches on the (short) user name entered for signin is: <code>roleFilter=member={0}</code> • The expression <code>{1}</code> denotes the distinguished name of the authenticated user as returned by the directory service (for example, <code>CN=john.doe@example.com,OU=Users,OU=Accounts,DC=example,DC=com</code> or <code>uid=john.doe@example.com,ou=People,o=example.com</code>). An example role filter that matches on the (full) authenticated user name is: <code>roleFilter=member={1}</code> <p>For more information, see Role Identification on page 135.</p>
uidAttributeID	<p>Identifies the group attribute that stores the directory service user ID.</p> <p>For example: <code>uidAttributeID=member</code></p> <p>For more information, see Role Identification on page 135.</p>
roleAttributeID	<ul style="list-style-type: none"> • For Active Directory, identifies the name of the user attribute that stores the user's group membership. For example: <code>roleAttributeID=memberOf</code> • For other directory services, identifies the name of the group attribute that stores the NNMi role that applies to the members of the directory server group. For example: <code>roleAttributeID=nnmiRole</code> <p>For more information, see Role Identification on page 135.</p>
roleAttributeIsDN	<p>Indicates how the user's role attribute is formatted in the directory service.</p> <ul style="list-style-type: none"> • For Active Directory, use: <code>roleAttributeIsDN=true</code> The group's role attribute (specified by <code>roleNameAttributeID</code>) represents the distinguished name of a role object, and the role name is taken from the value of the <code>roleAttributeID</code> attribute of the corresponding user object. • For other directory services, use: <code>roleAttributeIsDN=false</code> The role name is taken directly from the value of the group's role attribute (<code>roleAttributeID</code>). <p>For more information, see Role Identification on page 135.</p>

Table 10 Parameters in the ldap.properties File (cont'd)

Parameter	Description
roleNameAttributeID	(Active Directory) When <code>roleAttributeIsDN=true</code> , identifies the group attribute in the directory service that stores the NNMi role. An example usage is: <code>roleNameAttributeID=info</code>
userRoleFilterList	Optional. Limits the NNMi roles whose associated users can be assigned incidents in the NNMi console. The roles in this list apply only to directory service user names authenticated through LDAP. This parameter provides functionality that is not available when NNMi roles are assigned in the NNMi console and stored in the NNMi database. The format is a semicolon-separated list of one or more NNMi role names (as defined in Table 9 on page 135). For example: <code>userRoleFilterList=admin;level2;level1</code>
searchTimeLimit	Optional. Specifies the timeout value in milliseconds. The default value is 10000 (10 seconds). If you are encountering timeouts during NNMi user signin, increase this value. For example: <code>searchTimeLimit=10000</code>

Examples

Example
ldap.properties file
for Active Directory

An example `ldap.properties` file follows for Active Directory:

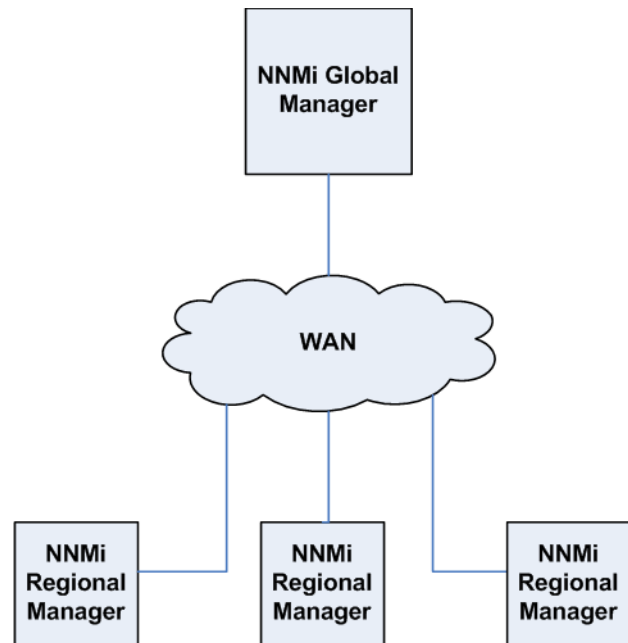
```
java.naming.provider.url=ldap://MYldapserver.example.com:389/
bindDN=MYdomain\MYusername
bindCredential=MYpassword
baseCtxDN=CN=Users,DC=MYldapserver,DC=EXAMPLE,DC=com
baseFilter=CN={0}
defaultRole=guest
rolesCtxDN=CN=Users,DC=MYldapserver,DC=EXAMPLE,DC=com
roleFilter=member={1}
uidAttributeID=member
roleAttributeIsDN=true
roleNameAttributeID=info
roleAttributeID=memberOf
userRoleFilterList=admin;level2;level1
```

Example
ldap.properties file
for other directory
services

An example `ldap.properties` file follows for other directory services:

```
java.naming.provider.url=ldap://MYldapserver.example.com:389/
baseCtxDN=ou=People,o=EXAMPLE.com
baseFilter=uid={0}
defaultRole=guest
rolesCtxDN=ou=Groups,o=EXAMPLE.com
roleFilter=member={1}
uidAttributeID=member
roleAttributeID=MYnnmiRole
userRoleFilterList=admin;level2;level1
```


Global Network Management



This chapter contains the following topics:

- Global Network Management Benefits
- Is Global Network Management a Good Tool for Managing my Network?
- Practical Global Network Management Examples
- Review the Requirements
- Initial Preparation
- Configuring Single Sign-On for Global Network Management
- Configuring Forwarding Filters on the Regional Managers
- Connecting a Global Manager with a Regional Manager
- Determining the Connection States from `global1` to `regional1` and `regional2`
- Reviewing `global1` Inventory
- Disconnecting Communication between `global1` and `regional1`
- Additional Information
- Configuring Application Failover for Global Network Management
- Troubleshooting Tips for Global Network Management
- Global Network Management and NNM iSPIs or Third-Party Integrations

Global Network Management Benefits

Suppose you have NNMi deployed on multiple NNMi management servers in several geographic locations. You have each NNMi management server discovering and monitoring the network to meet your discovery and monitoring needs. Using these existing NNMi management servers and configurations, you can designate specific NNMi management servers as global managers to display combined node object data without additional discovery or monitoring configuration changes.

The NNMi global network management feature enables multiple NNMi management servers to work together while managing different geographic areas of the network. You designate specific NNMi management servers as global managers to display combined node object data from 2 or more regional managers.

The NNMi global network management feature offers the following benefits:

- A central big-picture view of your corporate-wide network from the global manager.
- Easy to set up:
 - Each regional manager administrator specifies all node object data or a specific node group for participation at the global manager level.
 - Each global manager administrator specifies which regional managers are allowed to contribute information.
- Generates and manages incidents independently on each server (generated within the context of topology available on each server).

See *NNMi's Global Network Management Feature* in the NNMi help for additional details.

Is Global Network Management a Good Tool for Managing my Network?

Ask the following questions to determine if NNMi's global network management feature can help you better manage your network.

Do I Need Continuous Multi-Site Network Monitoring?

Does your information technology group manage network equipment located at multiple sites on a 24 by 7 basis? If so, your group can use NNMi's global network management feature to observe combined topology and incident views.

Can my Critical Devices be Visible?

From one NNMi management server, can I view device status and incidents for critical devices located at multiple locations? Yes. You configure forwarding filters on the regional managers. This enables you to select the node object data you want regional managers to send to global managers. For example, you can set up forwarding filters on the regional managers so that they only forward information about critical devices to the global manager.

Licensing Considerations

I currently have adequate NNMi licenses for single geographies. Can I use the global network management feature and limit the new licenses I need on the global manager? Yes. If your information technology group needs to monitor critical equipment located at multiple sites, you can configure a forwarding filter on the regional manager to

make sure you only forward information about critical devices to the global manager. This enables you to wisely use your NNMi investment and control your use of the license capacity you have on the global manager.

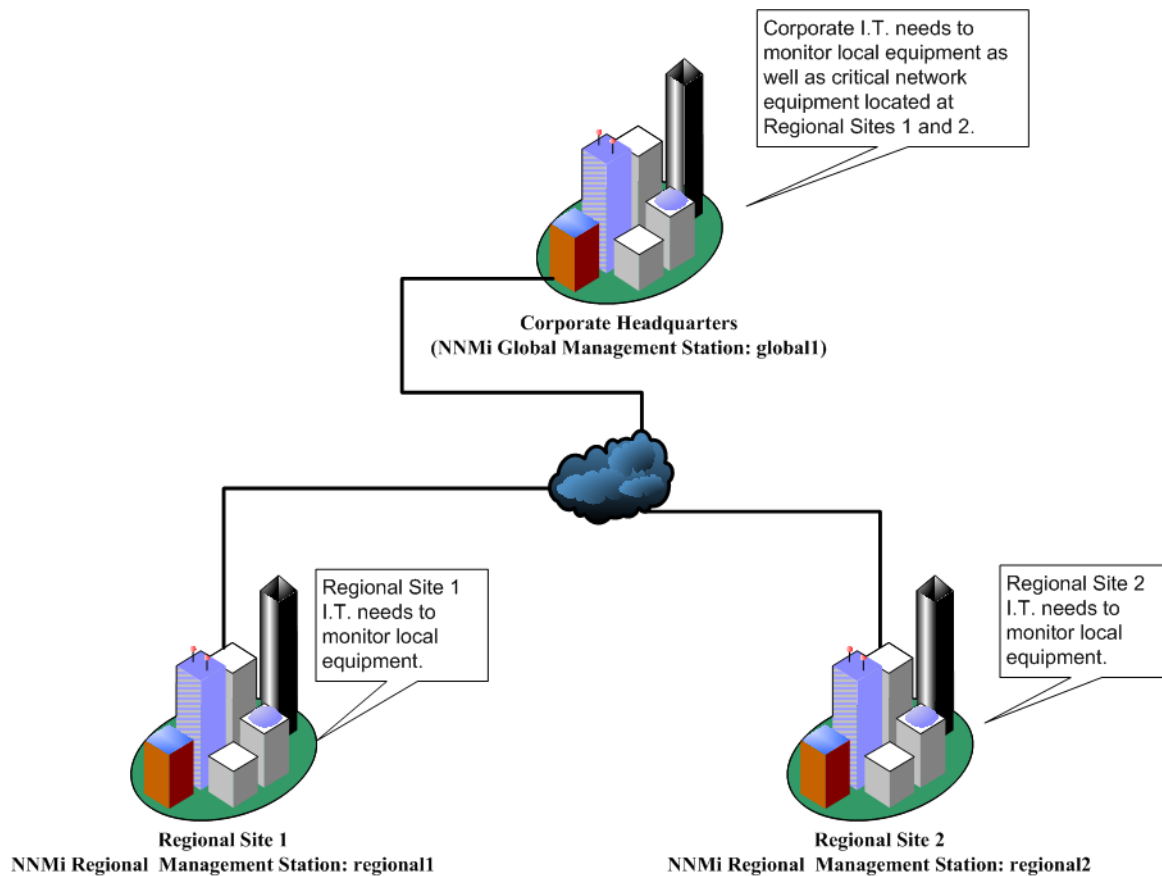
Do I need an NNMi Advanced license on both the global and regional managers? You must purchase and install an NNMi Advanced license on the NNMi management server you plan to use as a global manager. NNMi management servers do not need an NNMi Advanced license to function as regional managers.

Practical Global Network Management Examples

See [Figure 7](#) on page 147. Suppose your company has 2 operating sites in different geographic locations. Your company's headquarters is located in a third geographic area. There are NNMi management stations functioning at all 3 locations.

From a network perspective, information technologists located at corporate headquarters need to monitor local network equipment as well as critical network equipment located at both regional sites 1 and 2. Information technologists from both regional sites 1 and 2 need to monitor the local critical network equipment located at their sites.

Figure 7 Example Network.



Review the Requirements

Suppose the NNMi management servers at corporate headquarters, regional site 1, and regional site 2 manage several routers and switches located at their individual sites. For this example, refer to the NNMi management servers as `global1`, `regional1` and `regional2` respectively. Suppose you configured these NNMi management servers to discover and monitor critical switches and routers located at their own locations. There is no need to reconfigure discovery for NNMi management servers at any of these sites to use the global network management feature.



During global network management configuration, you might be tempted to use the `nnmbackup.ovpl` script to back up one NNMi management server, use the `nnmrestore.ovpl` script to restore this backup to a second NNMi management server, then connect both of these NNMi management servers to a regional NNMi management server. Do not do this. Placing the backup data from one NNMi management server onto a second NNMi management server means that both servers have the same database UUID. After you restore NNMi on the second NNMi management server, you would need to uninstall NNMi from the original NNMi management server.

The information technology group at your corporate site wants to monitor critical equipment located at regional sites 1 and 2, but they do not want to manage every device. The following table summarizes the monitoring needs:

Table 11 Network Requirements for Global Network Management

Site	NNMi Management Servers	Critical Switches	Regional Equipment to Manage
Corporate Headquarters	<code>global1</code>	15 Model 3500yl HP Procurve Switches	All model 3500yl HP Procurve Switches from each regional site
Regional Site 1	<code>regional1</code>	15 Model 3500yl HP Procurve Switches	not applicable
Regional Site 2	<code>regional2</code>	15 Model 3500yl HP Procurve Switches	not applicable

To summarize, you have an NNMi management server, `regional1`, monitoring the corporate headquarters. You have NNMi management servers, `regional1` and `regional2`, monitoring each of the regional sites. From corporate headquarters, you need to view incidents and device information for the Model 3500yl Procurve switches located at regional sites 1 and 2. Suppose that, for this example, `regional1` and `regional2` both manage several common switches located at regional site 1.

Regional Manager and Global Manager Connections

When you configure global network management connections, consider the following information:

- NNMi enables you to configure more than one global manager to communicate with a regional manager. For example, if you need a second global manager, `global2`, to communicate with `regional1`, NNMi enables you to configure both `global1` and `global2` to communicate with `regional1`. For more information see the *HP Network Node Manager i Software System and Device Support Matrix*.
- Global network management works with one connection layer. For example, the examples in this chapter discuss one connection layer: `global1` communicating with `regional1` and `global1` communicating with `regional2`. Do not configure NNMi for multiple connection levels. For example, do not configure `global1` to communicate with `regional1`, then configure `regional1` to communicate with `regional2`. The global network management feature is not designed for this three layer configuration.
- Do not configure two NNMi management servers to communicate both ways with each other. For example, do not configure `global1` to communicate with `regional1`, then configure `regional1` to communicate with `global1`.

Initial Preparation

Port Availability: Configuring the Firewall

For the global network management feature to function properly, you need to verify that certain well-known ports are open for TCP access from `global1` to `regional1`, and `regional2`. The NNMi installation script sets ports 80 and 443 as defaults; however, you can change these values during installation. Edit the following file to see the current values or to make changes:

- *Windows*: `%NNM_CONF%\nmm\props\nms-local.properties`
- *UNIX*: `$NNM_CONF/nmm/props/nms-local.properties`

The following table shows the well-know ports that need to be accessible:

Table 12 Required Accessible Sockets

	Security	Parameter	TCP Port
non-SSL		<code>jboss.http.port</code>	80
		<code>jboss.bisocket.port</code>	4457
		<code>jboss.jmsControl.port</code>	4458
SSL		<code>jboss.https.port</code>	443
		<code>jboss.sslbisocket.port</code>	4459
		<code>jboss.ssljmsControl.port</code>	4460

Configuring Self-Signed Certificates

If you plan to use the global network management feature with SSL (Secure Sockets Layer) between `global1` and the two regional NNMi management servers (`regional1` and `regional2`), you need to do some additional work. During NNMi installation, the NNMi installation script creates a self-signed certificate on the

NNMi management server so it can identify itself to other entities. You need to configure the NNMi management servers you plan to use with the global network management feature with the correct certificates. Complete the steps shown in [Configuring the Global Network Management Feature to use Self-Signed Certificates](#) on page 106.

Configuring Global Network Management for Application Failover

During NNMi installation, the NNMi installation script creates a self-signed certificate on the NNMi management server so it can identify itself to other entities. If you plan to use the application failover along with the global network management feature, you need to do some additional configuration. Complete the steps shown in [Configuring Global Network Management with Application Failover to use Self-Signed Certificates](#) on page 109.

NNMi Management Server Sizing Considerations

This example assumes you plan to use existing NNMi management servers in a global network management configuration. The global network management feature is different than the distributed solution used in earlier NNM products. The global network management feature avoids polling nodes being managed by regional systems, so you do not need to be as concerned about network bandwidth and computer resources.

Review the *NNMi 9.0 Installation Guide*, the *HP Network Node Manager i 9.0 Software Release Notes*, and the *HP Network Node Manager i Software System and Device Support Matrix*, for specific information about the size of server you need to house NNMi.

Synchronizing System Clocks

It is important for you to synchronize the NNMi management server clocks for `global1`, `regional1`, and `regional1` before you connect these servers in a global network management configuration. All NNMi management servers in your network environment that participate in global network management (global managers and regional managers) or single sign-on (SSO) must have their internal time clocks synchronized in universal time. Use a Time Synchronization program, for example, the UNIX (HP-UX / Linux / Solaris) tool Network Time Protocol Daemon (NTPD) or one of the available Windows operating system tools. See *Clock Synchronization Issues or Troubleshoot Global Network Management* in the NNMi help and [Clock Synchronization](#) on page 175 for more information.



NNMi displays a warning message at the bottom of the NNMi Console if there is a connection problem with a regional manager, such as a server clock synchronization problem.

Using the Application Failover Feature with Self-Signed Certificates in Global Network Management

If you plan to use the global network management feature using self-signed certificates in an application failover configuration, you need to complete some additional steps. See [Configuring Global Network Management with Application Failover to use Self-Signed Certificates](#) on page 109.

Using Self-Signed Certificates in Global Network Management

If you plan to use the global network management feature using self-signed certificates, you need to complete some additional steps. See [Configuring the Global Network Management Feature to use Self-Signed Certificates](#) on page 106.

Using a Certificate Authority in Global Network Management

If you plan to use the global network management feature using a Certificate Authority, you need to complete some additional steps. See [Configuring the Global Network Management Feature to use a Certificate Authority](#) on page 107.

List the Critical Equipment you Want to Monitor

Make a list of the equipment managed by `regional1` and `regional2` that you want to monitor from `global1`. You will use this information in a forwarding filter (to be discussed later). You need to carefully consider the possible outcomes of limiting the information forwarded to `global1` from `regional1` and `regional2`. Below are some things to consider during your planning:

- Be careful not to exclude too many devices, as `global1` needs a complete topology from `regional1` and `regional2` to do a complete analysis to generate accurate incidents.
- Excluding non-critical devices helps you to reduce license costs on `global1`.
- Excluding non-critical devices helps you to improve the solution's overall scalability, and reduce the network traffic required by NNMi.

Review the Global and Regional Managers' Management Domains

NNMi management servers `global1`, `regional1`, and `regional2` manage their own set of nodes. Later in this example, you will configure `regional1` and `regional2` to forward information about equipment they manage to `global1`.

Use the following procedure to understand the equipment that `global1`, `regional1`, and `regional2` currently monitor. This helps you select the critical equipment you want `regional1` and `regional2` to forward to `global1`.

For this example, complete the following steps to review this information:

- 1 Point your browser to `global1`'s, NNMi console.
- 2 Sign in.
- 3 Click **Inventory** workspace.
- 4 From here you can review the discovered inventory `global1` currently monitors.
- 5 Point your browser to `regional1`'s, NNMi console.
- 6 Sign in.
- 7 Click **Inventory** workspace.
- 8 Review the nodes that `regional1` monitors and make a list of the devices you want to monitor from `global1`.
- 9 Point your browser to `regional2`'s, NNMi console.
- 10 Sign in.
- 11 Click **Inventory** workspace.

- 12 Review the nodes that `regional2` monitors and make a list of the devices you want to monitor from `global1`.

Review NNMi Help Topics

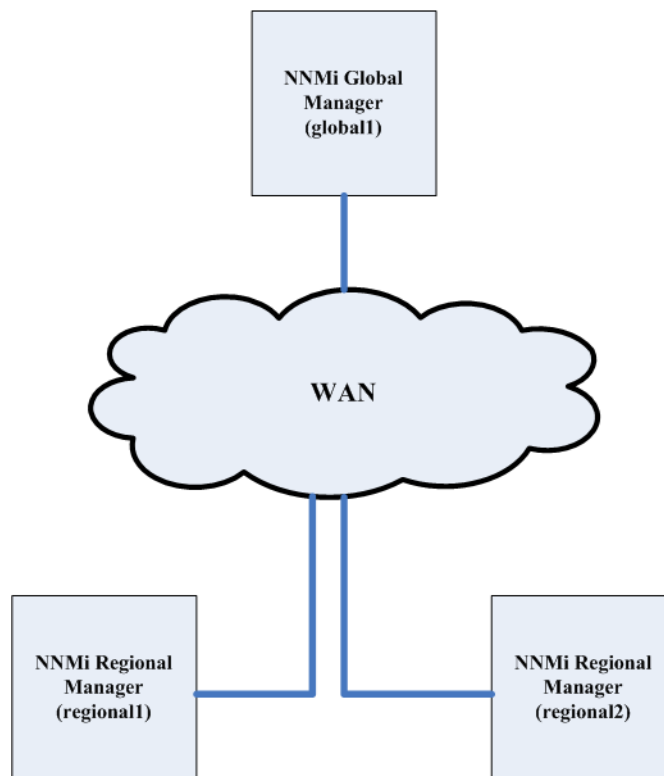
To review all of the help topics related to global network management, complete the following steps:

- 1 From the NNMi help, click **Search**.
- 2 Type **"Global Network Management"** in the `Search` field.
- 3 Click **Search**.

This search results in more than 50 topics related to global network management.

Configuring Single Sign-On for Global Network Management

You can configure SSO (HP Network Node Manager i Software single sign-on) to facilitate access to NNMi regional managers from an NNMi global manager. You need to complete this step before connecting regional managers from a global manager. See [Using Single Sign-On with NNMi](#) on page 113 for more information.



The SSO feature communicates a user name among NNMi management servers, but not passwords or roles. For example, NNMi associates the same username on one NNMi management server (`global1`) with a different role on other NNMi management servers (`regional1` or `regional2`). Any of these three NNMi management servers could associate a different password with the same username.

If a global and regional manager resides in the same management domain, and you do not copy the *Initialization String* value from the global NNMi management server to the regional NNMi management server as shown in [step 3](#) on page 153, you could have NNMi console access problems. To avoid this, either configure SSO correctly using the following steps, or disable SSO as shown in [Disabling SSO](#) on page 116.

To configure SSO to work with the global network management feature, complete the following steps:

- 1 Locate the SSO NNMi initialization string for `global1`. Look in the following locations:
 - *Windows*: `%NNM_SHARED_CONF%/lwssofmconf.xml`
 - *UNIX*: `$NNM_SHARED_CONF/lwssofmconf.xml`
- 2 Open the `lwssofmconf.xml` file from `global1` and look for a section in the file that resembles the following:

```
<lwssValidation id="ID000001">
  <domain>cnd.hp.com</domain>
  <crypto cipherType="symmetricBlockCipher"
    engineName="AES" paddingModeName="CBC" keySize="256"
    encodingMode="Base64Url"
    initString=Initialization String></crypto>
</lwssValidation>
```

- 3 Copy the real value of *Initialization String* from the `lwssofmconf.xml` file on `global1` to the `lwssofmconf.xml` files on `regional1` and `regional2`. All of the servers must use the same value for *Initialization String*. Save your changes.



NNMi supports copying the *Initialization String* value from the global NNMi management server to the regional NNMi management servers. In this step, you copied the *Initialization String* value from the global manager to the two regional managers. Always copy the *Initialization String* value from the global manager to the regional managers if you want to use SSO with the global network management feature.



If a global and regional manager resides in the same management domain, and you do not copy the *Initialization String* value from the global NNMi management server to the regional NNMi management server, disable SSO to avoid NNMi console access problems. See [Disabling SSO](#) on page 116 for more information.

- 4 If `global1`, `regional1`, and `regional2` are in different domains, modify the `protectedDomains` content. To do this, look in the `lwssofmconf.xml` file for a section that resembles the following:

```
<protectedDomains>
  <url>xxx.hp.com</url>
</protectedDomains>
```

- a Suppose `global1` is in `country1.hp.com`, `regional1` is in `country2.hp.com` and `regional2` is in `country3.hp.com`. Modify the `protectedDomains` section of the `lwssofmconf.xml` file on `global1`, `regional1` and `regional2` as follows:

```
<protectedDomains>
```

```
<url>hp.com</url>
</protectedDomains>
```

Save your changes.

- b** Suppose `global1` is in `colorado.hp.com`, `regional1` is in `california.mycompany.com`, and `regional2` is in `hawaii.yourcompany.com`. Modify the `protectedDomains` section of the `lwssofmconf.xml` file on `global1`, `regional1`, and `regional2` as follows:

```
<protectedDomains>
  <url>colorado.hp.com</url>
  <url>california.mycompany.com</url>
  <url>hawaii.yourcompany.com</url>
</protectedDomains>
```

Save your changes.

- 5** Run the following command sequence on servers A, B, and C:

- a** **ovstop ovjboss**
- b** **ovstart ovjboss**



There are no manual configuration steps to perform to enable single sign-on in an application failover configuration. For example, If you plan to configure single sign-on in an application failover configuration, NNMi replicates the above changes from the active NNMi management server to the standby NNMi management server.

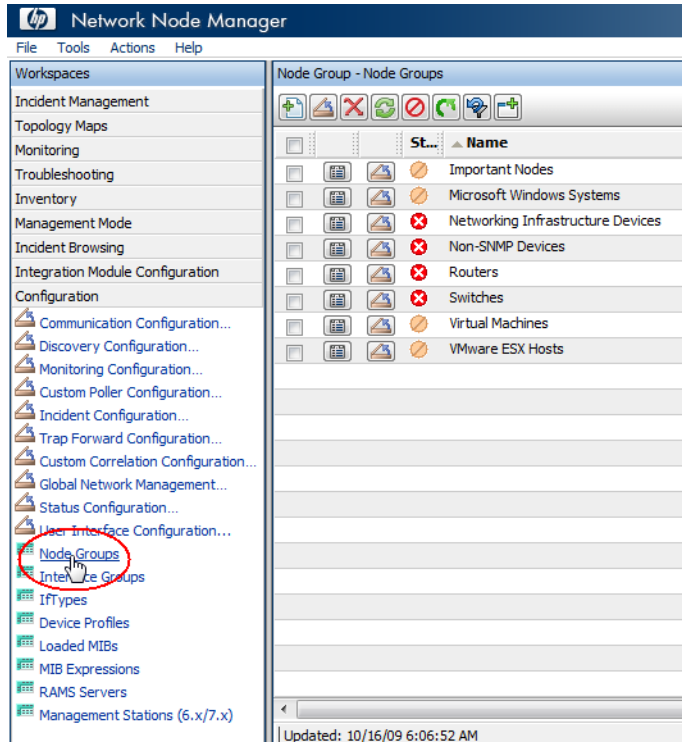
Configuring Forwarding Filters on the Regional Managers

In this example, `global1` communicates with both `regional1` and `regional2`. To control the node object data you want the global manager, `global1`, to receive from regional managers `regional1` and `regional2`, you need to configure forwarding filters on both `regional1` and `regional2`.

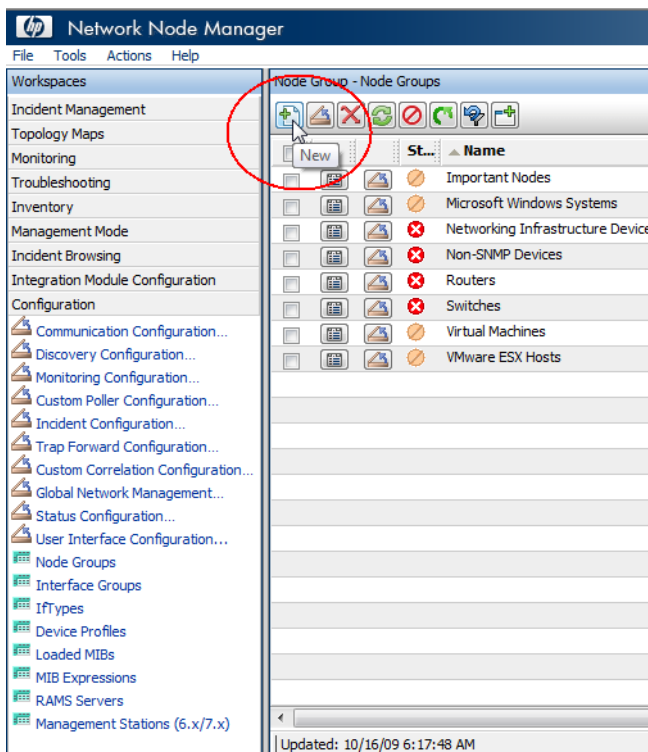
Configuring a Forwarding Filter to Limit Forwarded Nodes

Suppose you want to set up a node group to enable `regional1` to only forward node information for Procurve Model 3500yl switches to `global1`. To create a new node group and set these limits, complete the following steps:

- 1** From `regional1`'s Configuration workspace in the NNMi console, click **Node Groups**.



2 Click **New**.



▶ Although this example explains how to create a new node filter, then use it to create a forwarding filter from regional1 and regional2, you can use any of these existing filters to set up forwarding filters from a regional NNMi management server to a global NNMi management server.

▶ You can create a *container* node group that contains no devices or filters of its own; then use this node group to specify child node groups. Using this approach, you can forward node object data to global NNMi management servers using one *container* node group.

- 3 Click the **Device Filters** tab. Type `global1` as the filter name and make any notes you need about the filter you are creating in the notes field.

File View Tools Actions Help

Save and Close Delete Node Group

NNMi's self monitoring has detected a problem (Minor). Please see Help → System Information → Health for details.

Basics

Name: global1

Status: No Status

Add to View Filter List:

Notes: Pass Procurve Switches to global1

You can filter Node Groups using Device Filters and Additional Filters. A node must pass all of these filters to belong to a specified Node Group. Any node that is listed as an additional node or is a member of a Child Node Group will belong to this Node Group. See Help → Using the Node Group form.

To test your Node Group definition, select File → Save, then Actions → Show Members.

NNM iSPI Performance

Used by NNM iSPI Performance for Metrics and NNM iSPI Performance for Traffic.

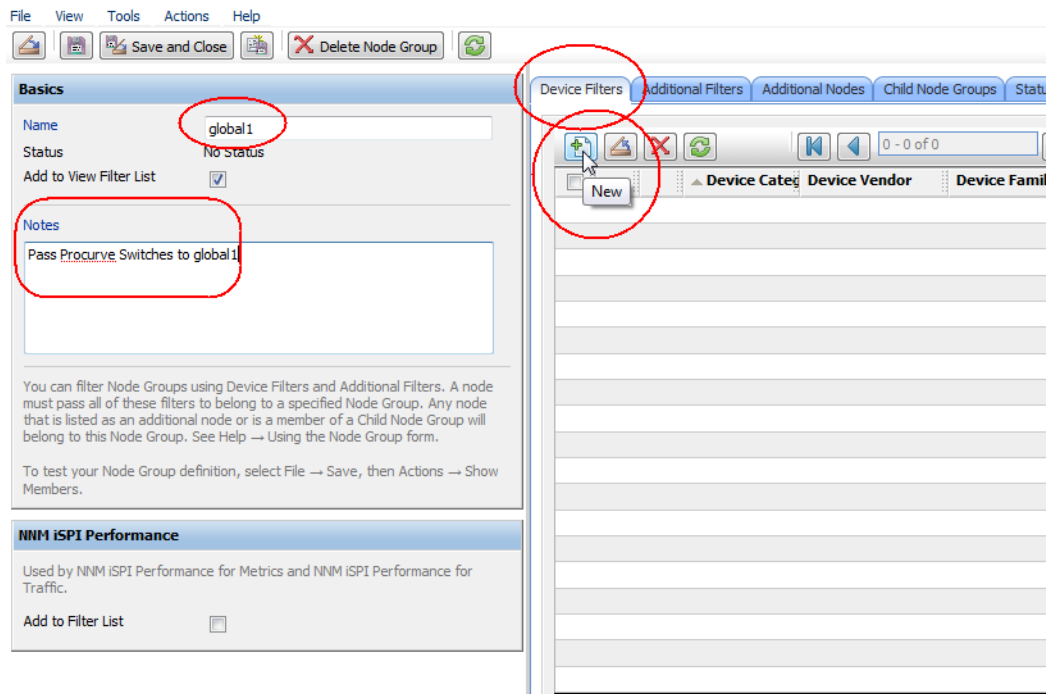
Add to Filter List:

Device Filters Additional Filters Additional Nodes Child Node Groups Status

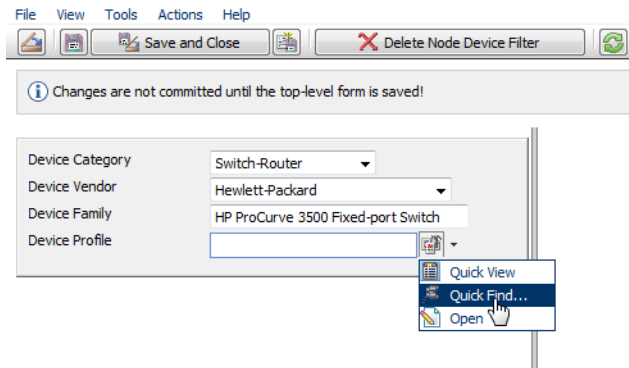
Device Cat...	Device Vendor	Device Family
---------------	---------------	---------------

Updated: 10/16/09 4:17:00 | Total: 0 | Selected: 0 | Filter: OFF | Auto refre

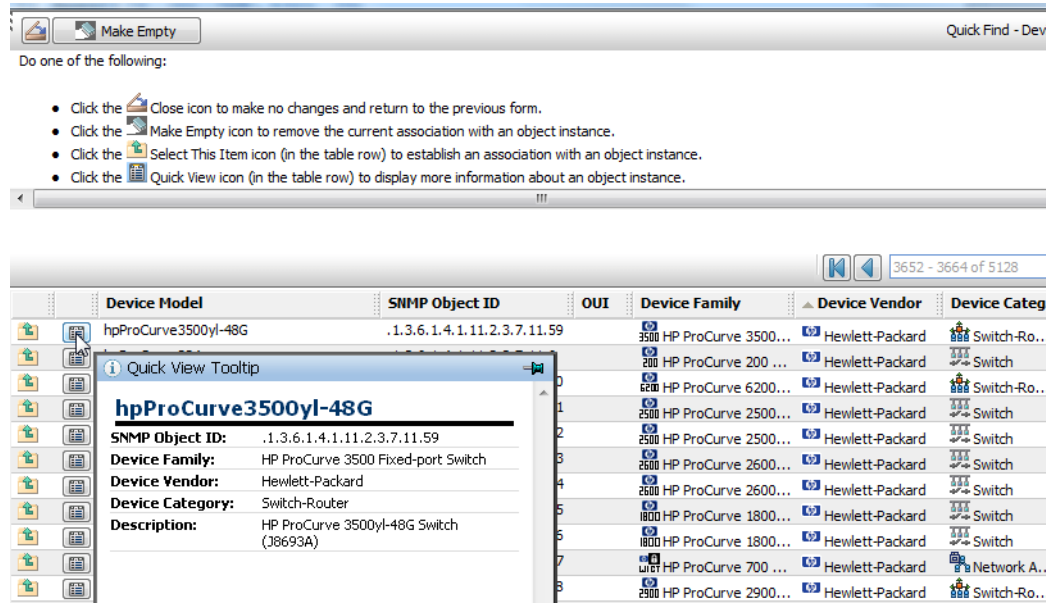
- 4 Click the **New** icon to open a Node Device Filter form.



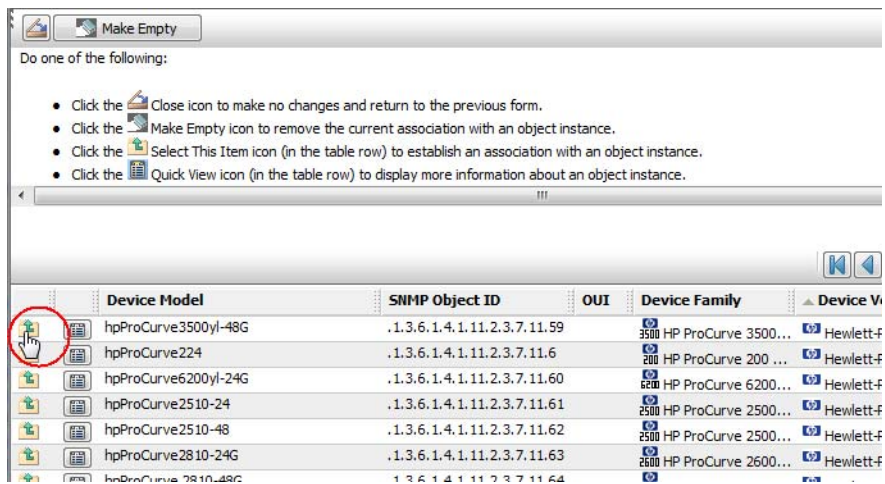
- 5 Using the pull-down, select the **Switch Router Device Category**, the **Hewlett-Packard Device Vendor**, and the **HP Procurve 3500 Fixed-port Switch Device Family**.
- 6 Using the pull-down, click **Quick Find** to open a Device Profile form.



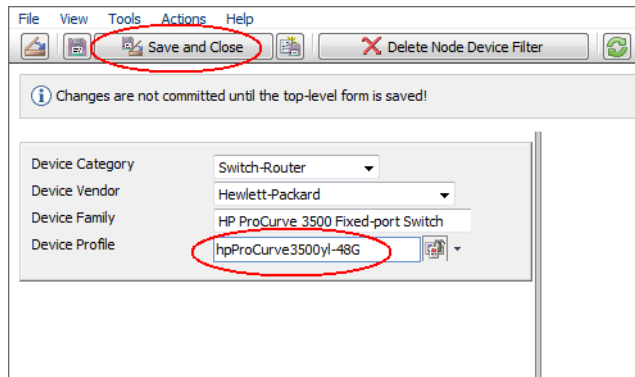
7 Find the profile for the HP Procurve 3500yl Switch.



8 Select the profile for the HP Procurve 3500yl Switch.

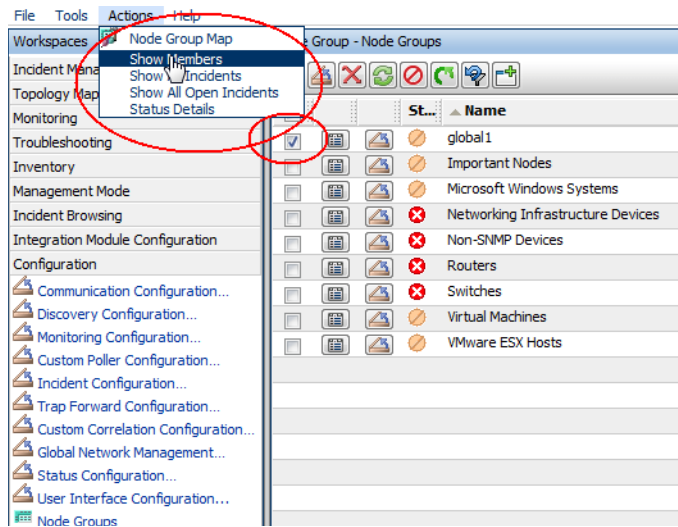


9 Click **Save and Close** two times.

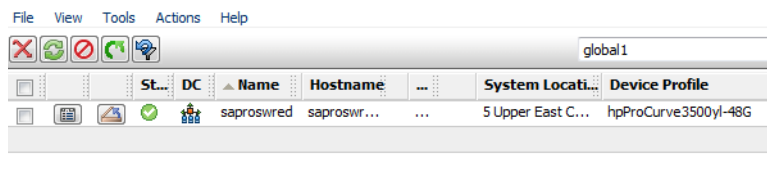


10 To test this filter, select **global1**.

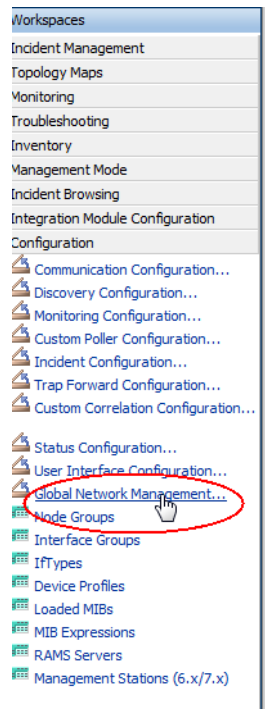
11 Using the pull-down, click **Show Members**.



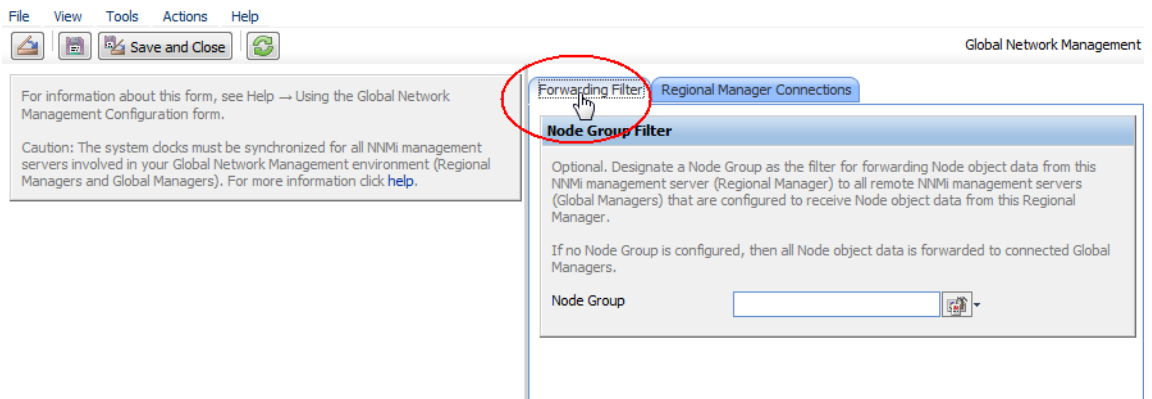
12 Notice that NNMi discovered 1 HP 3500yl switch already. This shows you that the filter you created is finding the specific switch models you configured it for. The next step is to configure the forwarding filter using this node filter you just created.



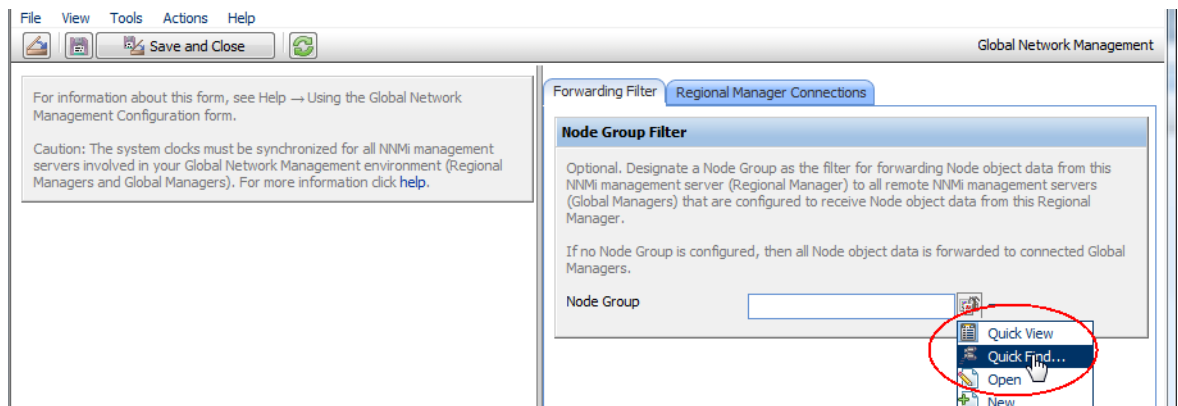
13 From regional1's Configuration workspace in the NNMI console, click **Global Network Management**.



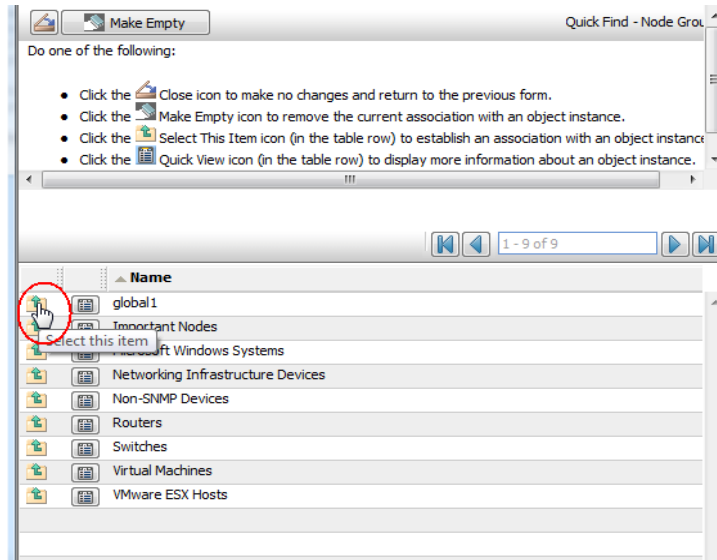
14 Click the **Forwarding Filter** tab.



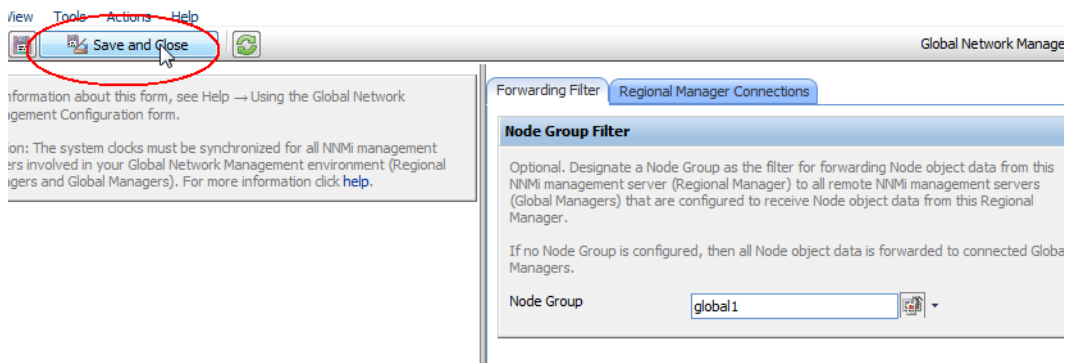
15 Click **Quick Find**.



16 Click the **global1** filter.



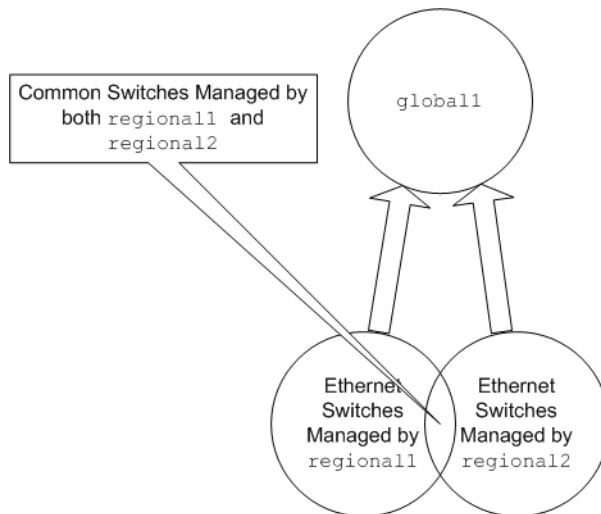
17 Click **Save and Close**.



This completes the task of setting up a forwarding filter on regional1. After you complete [step 1](#) through [step 17](#) for regional2, move on to the next section to connect global1 to regional1 and regional2.

Connecting a Global Manager with a Regional Manager

As mentioned earlier, suppose that `regional1` and `regional2` both manage several common switches. Suppose that you want this common switch information forwarded to `global1` from `regional1`.



To make that happen you must connect `global1` to `regional1` before connecting it to `regional2`. By using that connection sequence, `global1` considers `regional1` to be the NNMi management server monitoring these common switches. `Global1` also ignores information about these common switches that it receives from `regional2`.

▶ HP recommends you use this feature on a small scale to better understand how it works, then expand it to meet your network management needs.

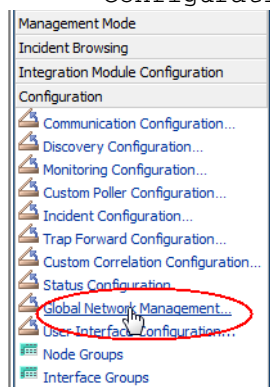
To connect `global1` first to `regional1`, then to `regional2`, complete the following steps:

- 1 As mentioned earlier, synchronize the NNMi management server clocks for `global1`, `regional1`, and `regional2` before you connect these servers in a global network management configuration. See *Clock Synchronization Issues* in the NNMi help for more information.

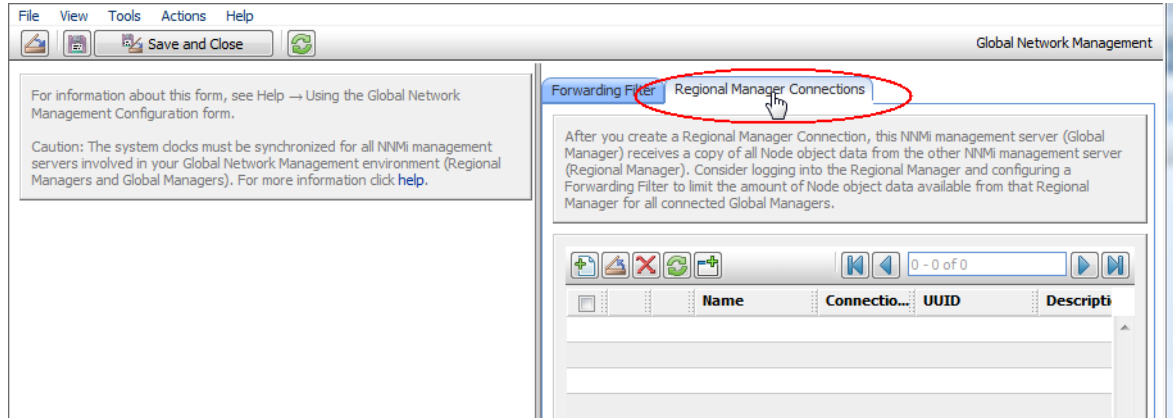
▶ NNMi displays a warning message if there is a connection problem with a regional manager, such as a server clock synchronization problem.


- 2 Set up a connection from `global1` to `regional1`.

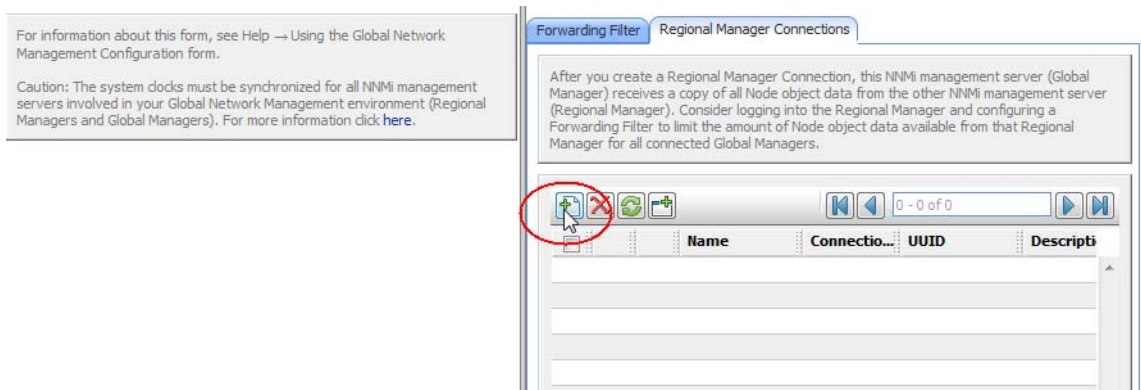
- a From the `global1` NNMi console, click **Global Network Management** in the Configuration workspace.



b Click **Regional Manager Connections**.




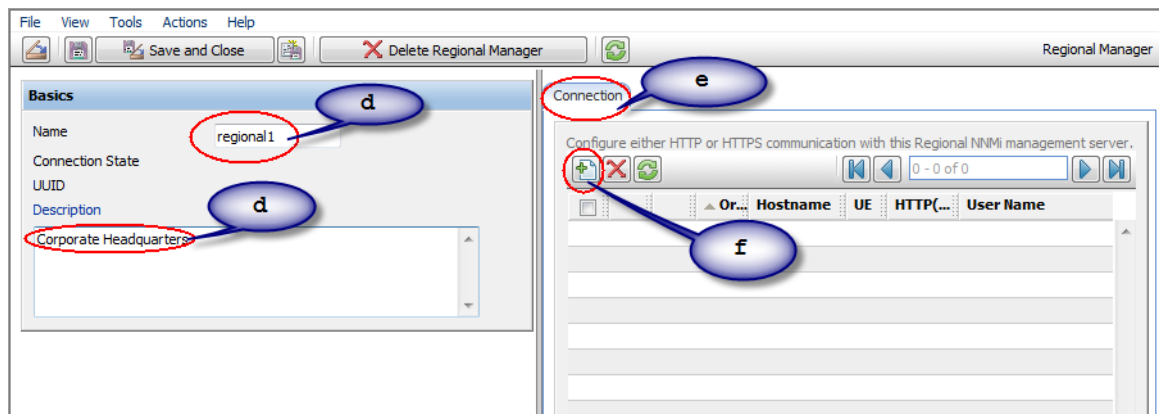
c Click the  icon to create a new regional manager.



d Add the name and description information for regional1.

e Click the **Connection** Tab.

f Click the  icon.



g Add the connection information for regional1



See **Help->Using the Regional Manager Connection Form** in the NNMi help for specific information about the entries to make in this form

File View Tools Actions Help

Save and Close Delete Regional Manager Connection Regional Manager Connectio

Changes are not committed until the top-level form is saved!

Provide the fully-qualified domain name for the hostname of the remote Regional Manager server. See Help → Using the Regional Manager Connection form for more information.

Hostname regional1

Use Encryption

HTTP(S) Port

User Name system

User Password

Ordering

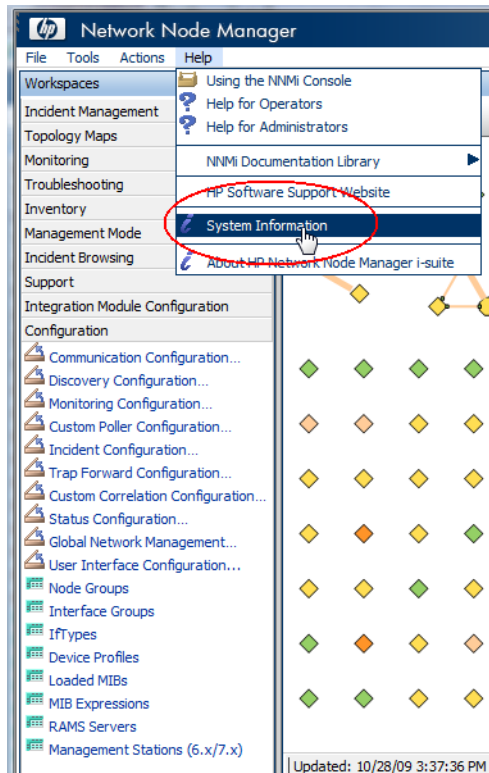
h Click **Save and Close** two times to save your work.

3 Complete [step a](#) on page 162 through [step g](#) on page 164 to establish a connection from global1 to regional2.

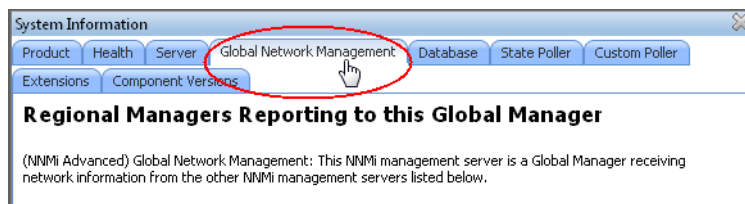
Determining the Connection States from global1 to regional1 and regional2

To check the connection states from global1 to regional1 and regional2, complete the following steps:

- 1 From the global1 NNMI console, click **Help > System Information**

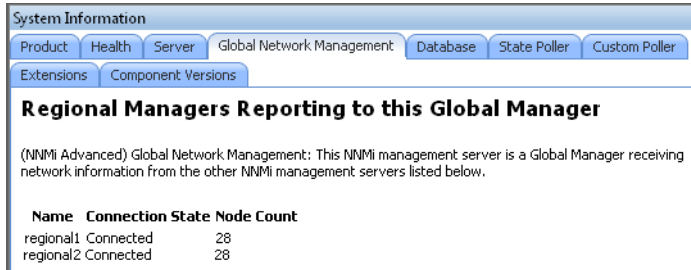


- 2 Click the **Global Network Management** tab.



- 3 Check the status of `regional1` and `regional2` by checking their connection states. Notice that the connection states are shown as `Connected`, which means they are functioning properly.

See *Determine the State of the Connection to a Regional Manager* in the NNMi help for more information.



The screenshot shows the 'System Information' page with the 'Global Network Management' tab selected. Under the 'Component Versions' sub-tab, the section 'Regional Managers Reporting to this Global Manager' is displayed. It includes a descriptive paragraph and a table with the following data:

Name	Connection State	Node Count
regional1	Connected	28
regional2	Connected	28

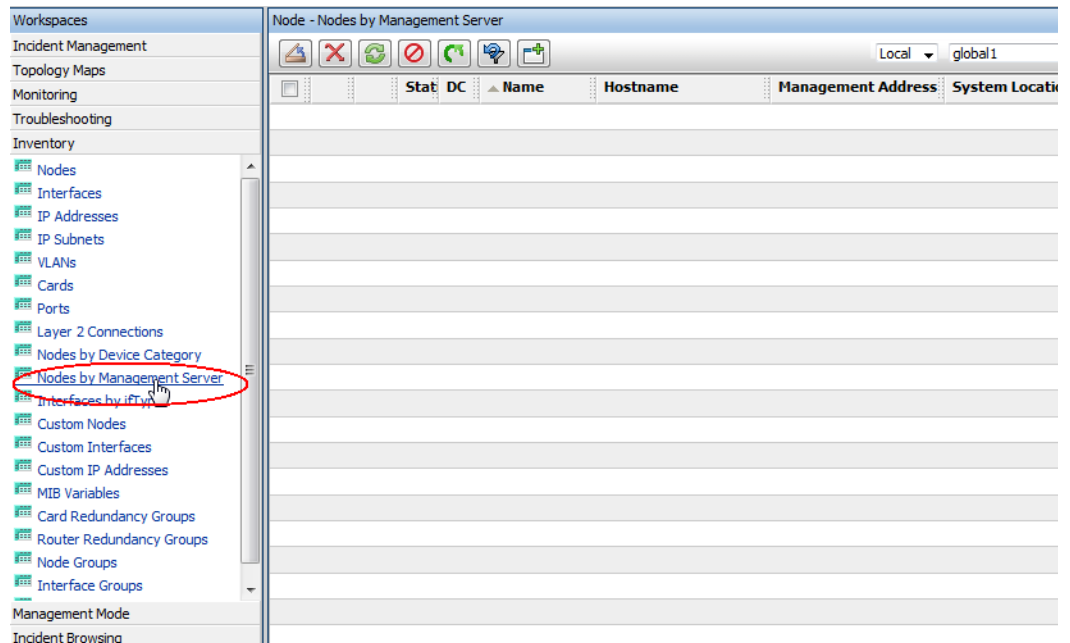
Do not continue to the next section until NNMi completes a good discovery. See *Checking Discovery Progress* in the *NNMi Installation Guide* for more information.

Reviewing global1 Inventory

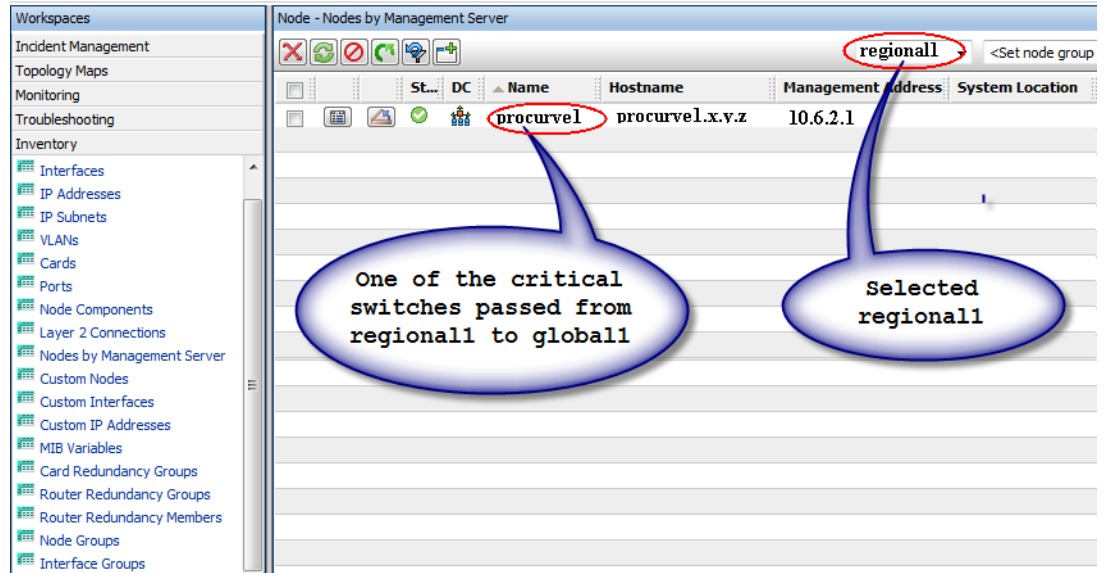
Do not complete this section until NNMi completes a good discovery. See *Checking Discovery Progress* in the *NNMi Installation Guide* for more information.

To view the node information `regional1` forwarded to `global1`, complete the following steps:

- 1 From the `global1` NNMi console, navigate to the `Nodes by Management server` form located in the `Inventory` workspace.



- 2 Assume that regional1 passed information about switch `procurvel.x.y.z` to `global1`. After you select **regional1**, the inventory might look as follows:

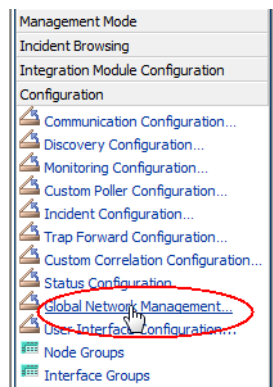


Complete [step 1](#) through [step 2](#) to look at the device inventory passed to `global1` from other regional managers you connected it with.

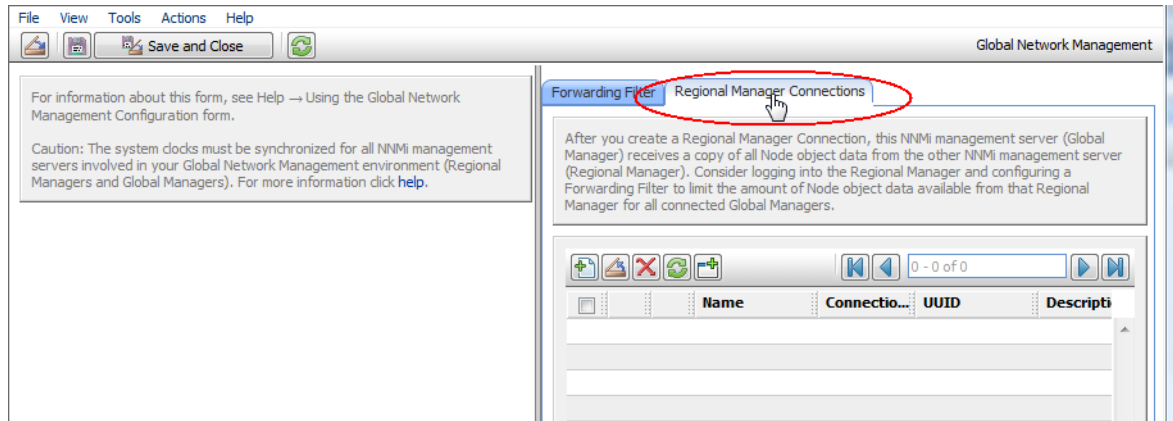
Disconnecting Communication between `global1` and `regional1`

Suppose you plan to permanently shut down `global1` or to shut it down for a number of days. Assume for this example that `global1` still has active subscriptions to `regional1`. You need to complete some additional steps to complete the shutdown:

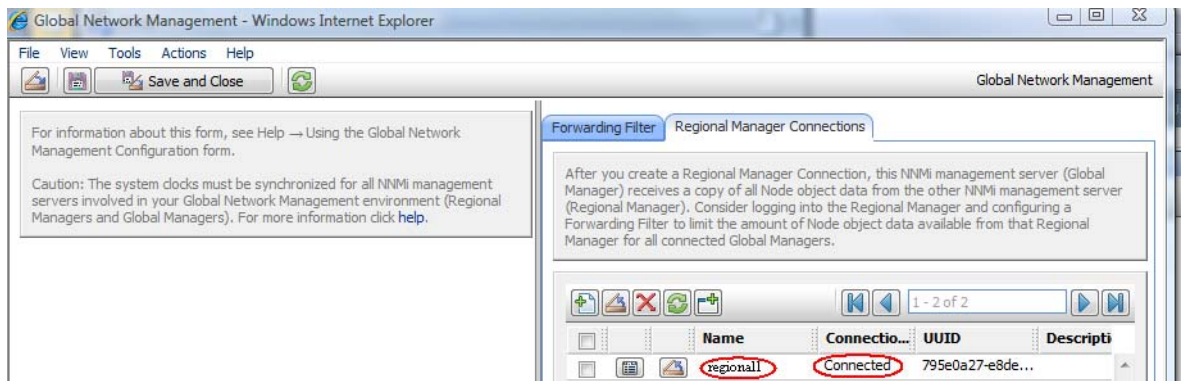
- 1 From the `global1` NNMi console, click **Global Network Management** in the Configuration workspace.



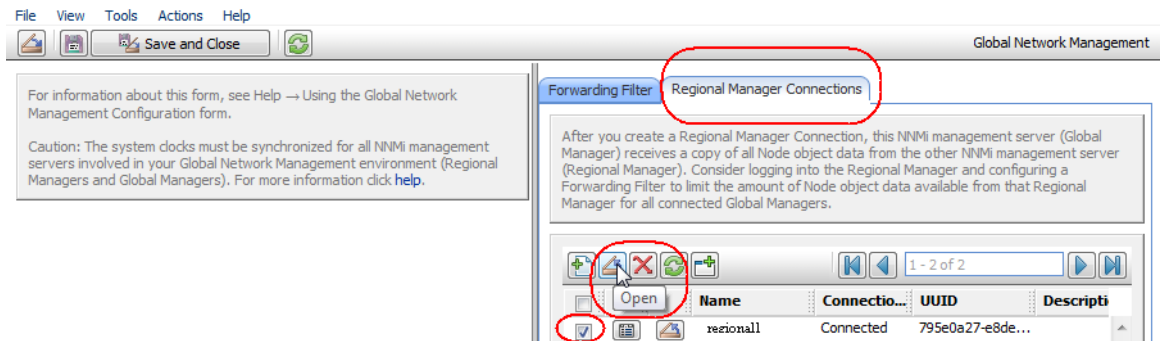
2 Click Regional Manager Connections.



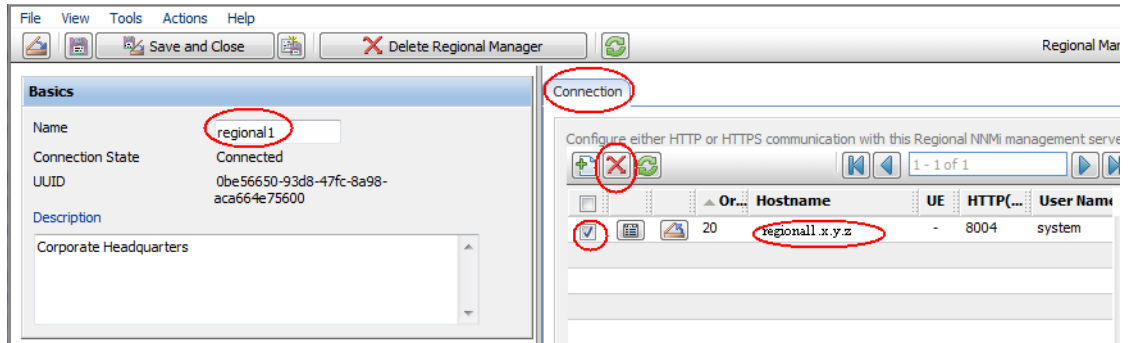
3 Check to make sure the status is Connected. If the status is not Connected, diagnose the problem using information from the *Troubleshoot Global Network Management* topic in the NNMI help before continuing.



4 Select regional1, then click the Open icon.

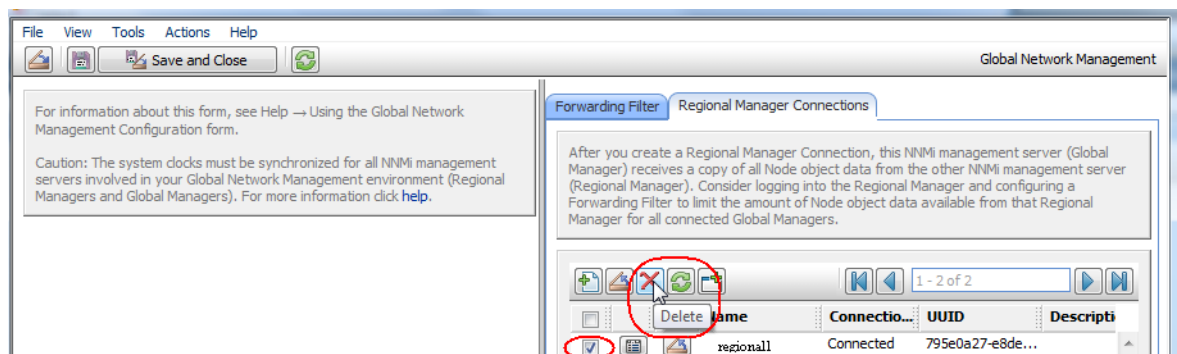


- 5 Click **Connection**, select **regional1.x.y.z**, then click the **Delete** icon.



- 6 Click **Save and Close**.
- 7 In the **Regional Manager Connections** tab, note the **Name** attribute value for **regional1** (case-sensitive). You need this text string for the **RemoteNNMiServerName** variable in a later step.
- 8 Click **Save and Close** again.
- 9 On **global1**, at the command line, type the following command:


```
nnmnodedelete.ovpl -rm regional1 -u NNMiadminUserName -p NNMiadminPassword
```
- 10 These commands remove the node records from **global1** that **regional1** forwarded to it. The commands also close incidents associated with the nodes forwarded to **global1** from **regional1**. For detailed information, see *Disconnect Communication with a Regional Manager* in the NNMi help.
- 11 To remove the configuration records for **regional1**, do the following.
 - a From the workspace navigation panel, click the **Configuration** workspace.
 - b Select the **Global Network Management** form.
 - c Select the **Regional Manager Connections** tab.
 - d Select **regional1**, then click the **Delete** icon.



- e Click **Save and Close** to save your deletions.
- 12 Complete [step 1](#) through [step 11](#) for other regional NNMi management servers, such as **regional2**, that are connected to **global1**.

Discovery and Data Synchronization

As network administrators add, delete, or modify devices on a network, regional servers, such as `regional1` and `regional2`, discover those changes and update global servers, such as `global1` in the example in this chapter, `regional1` and `regional2` also notify `global1` of changes that administrators make to the management mode of a node it manages.



To maintain consistency, as `regional1` and `regional2` discover device state changes, they continuously update `global1`, thereby maintaining identical node states on both the global and regional servers.

Any time `global1` requests information about a node that is managed by `regional1` or `regional2`, `regional1` or `regional2` responds to `global1` with the requested information. `global1` never talks directory to a node. You do not introduce discovery process duplication by querying for node information from `global1` when the node is managed by either `regional1` or `regional2`.

`global1` synchronizes with `regional1` and `regional2` each time `regional1` or `regional2` completes a discovery. NNMi uses FDB (Forwarding Database) data to calculate layer 2 connections. FDB data is very dynamic, and varies a lot between synchronizations, especially if there are multiple regionals connected to a global. A synchronization does the following important calculations and updates:

- NNMi updates the FDB, DOT1DPORT, IFACE_STACK, and VLAN data.
- NNMi completes a layer 2 FDB connection analysis

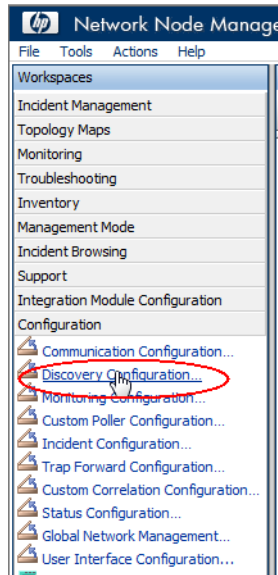
The above list is not intended to be a complete list, but shows some of the main data that NNMi updates during a synchronization.



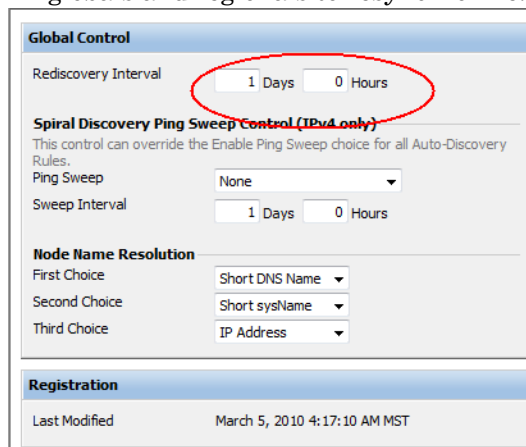
Changes to user-modified or application-modified attributes are not updated on the global during a synchronization.

The `Rediscovery Interval` is adjustable on each regional, and can make a difference in the discovery accuracy between `global1` and the regional managers. The shorter the `Rediscovery Interval`, the more accurate the discovery, and the more NNMi-generated network traffic. The longer the `Rediscovery Interval`, the less accurate the discovery, and the less NNMi-generated network traffic. This means that the larger your network grows, the less frequently you may want to rediscover. To set the `Rediscovery Interval`, do the following steps:

- 1 From the global NNMI console, click **Discovery Configuration** in the Configuration workspace.



- 2 Adjust the Rediscovery Interval time according to your how often you want the globals and regionals to resynchronize.

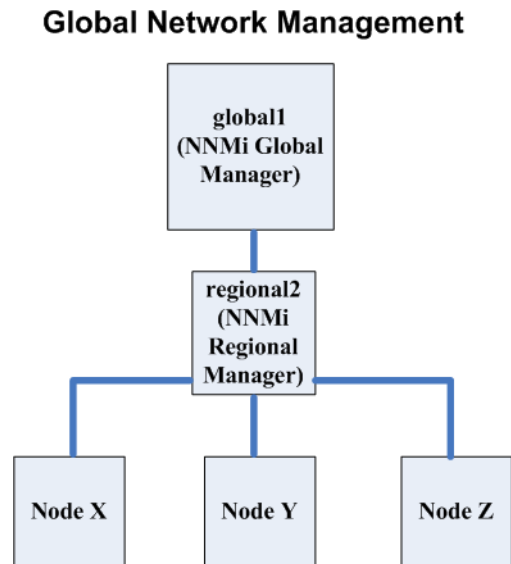


- 3 Click **Save and Close**.

Status Poll or Configuration Poll a Device

Suppose regional NNMi management server `regional2` discovers and manages Node X and global NNMi management server `global1` connects with regional NNMi management server `regional2`.

Figure 8 Status Poll or Configuration Poll a Node



To status poll Node X from `global1`, do the following:

- 1 From `global1`, click **Nodes** in the **Inventory** workspace.
- 2 Select Node X from the nodes inventory.
- 3 Request a status poll of Node X using the **Actions > Status Poll** menu item.
- 4 NNMi management server `global1` requests a status poll from regional NNMi management server `regional2` and shows the results on your screen. It does not matter if you initiate the status poll request from either `global1` or `regional2`. You still see the same status poll results.

If you want `global1` to have the most current discovery information for Node X, do the following to configuration poll Node X from `global1`.

- 1 From `global1`, click **Nodes** in the **Inventory** workspace.
- 2 Select Node X from the nodes inventory.
- 3 Request a configuration poll of Node X using the **Actions > Configuration Poll** menu item.
- 4 NNMi management server `global1` requests a configuration poll from regional NNMi management server `regional2` and shows the results on your screen. It does not matter if you initiate the configuration poll request from either `global1` or `regional2`. You still see the same configuration poll results.

Determining Device Status and NNMi Incident Generation using a Global Manager

NNMi management server `global1` listens for state changes coming from regional managers `regional1` and `regional2` and updates the states in its local database.

The NNMi `StatePoller` services on NNMi management servers `regional1` and `regional2` calculate state values for the devices it monitors. `global1` receives state value updates from `regional1` and `regional2`. `global1` polls nodes that it discovers, and does not poll nodes being managed by `regional1` and `regional2`.

After you change the management mode of a node being managed by `regional1`, you see that management mode change on `global1` as well. As network administrators add, remove, or modify network equipment being managed by `regional1` or `regional2`, `regional1` or `regional2` updates `global1` of these network device changes.

`global1` generates incidents using its own causal engine and topology, including the node object data forwarded to it by `regional1` and `regional2`. This means that the incidents it generates may be slightly different from the `regional1` and `regional2` incidents if there are differences in topology.

It is better to avoid using a forwarding filter on `regional1` or `regional2`, as filtering may affect the connectivity on `global1`. The result could be a difference in the root cause analysis between `global1` and the two regionals (`regional1` and `regional2`). In most cases, if you choose to avoid using forwarding filters, a global NNMi management server will have a larger topology. This helps it draw more accurate root cause analysis conclusions.

Without additional configuration, `regional1` does not forward traps to `global1`. To do this, you must configure `regional1` to forward specific traps to `global1`. HP recommends you only configure regional managers to forward low-volume, important traps to avoid excessive burden on the global manager. NNMi drops forwarded traps if the forwarded traps result in a `TrapStorm` incident. See the `TrapStorm` Management Event details in the NNMi console.

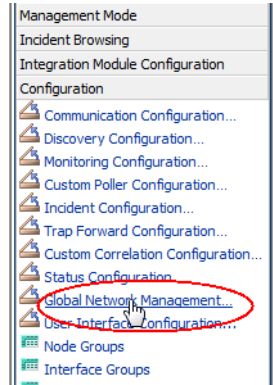
Configuring Application Failover for Global Network Management

You can configure both global and regional managers to use application failover. The global or regional manager automatically detects and connects to the active system.

Configuring Application Failover on a Global Manager

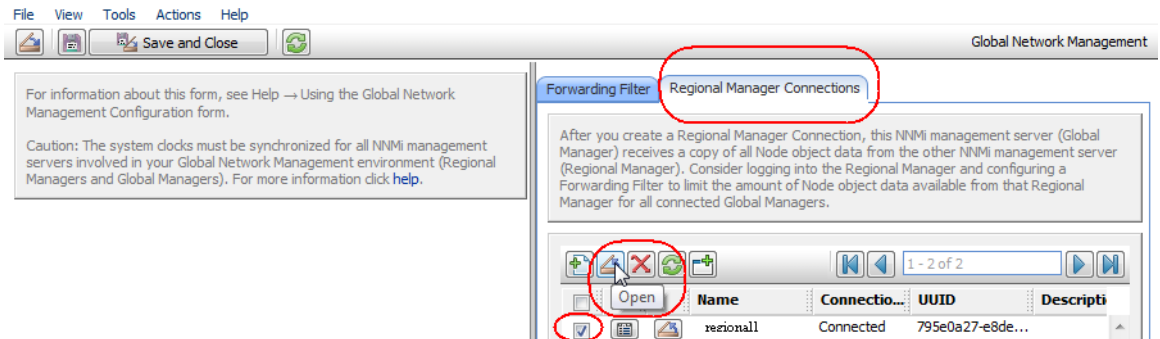
To configure `global1` to recognize the application failover do the following:

- 1 From the `global1` NNMI console, click **Global Network Management** in the **Configuration** workspace.

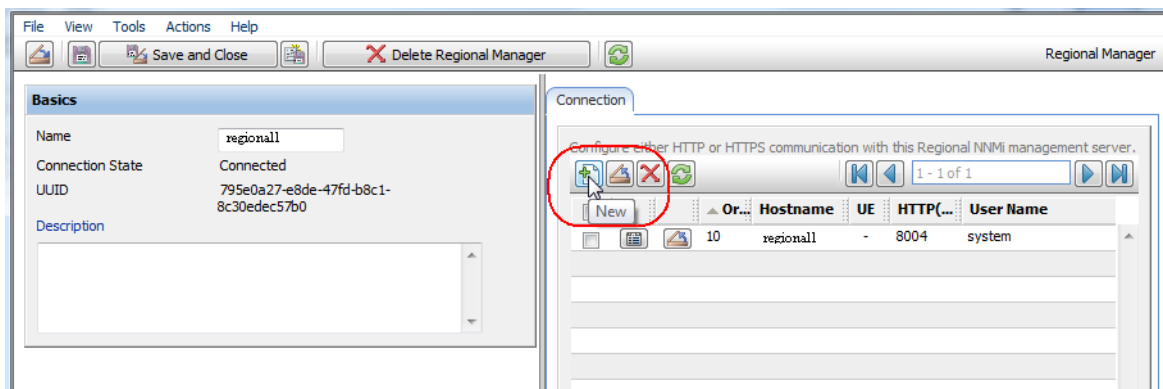


Suppose you configured `regional1` for application failover, and `regional1_backup` as the secondary server.

- 2 Click **Regional Manager Connections**.
- 3 Select `regional1`, then click the **Open** icon.



- 4 Click the **New** icon.



- 5 Add the **Hostname**, **HTTP(S) Port**, **User Name** and **Ordering** value. Set the ordering value to a value greater than the regional1 value.

File View Tools Actions Help

Save and Close Delete Regional Manager Connection

Changes are not committed until the top-level form is saved!

Provide the fully-qualified domain name for the hostname of the remote Regional Manager server. See Help → Using the Regional Manager Connection form for more information.

Hostname: regional1_backup.x.y.z

Use Encryption:

HTTP(S) Port: 80

User Name: system

User Password:

Ordering: 200

- 6 Click **Save and Close** three times to save your work.

If a regional manager fails, the global manager does the following:

- a It contacts the primary.
- b If the primary does not respond, it contacts the secondary.

If the global system detects that the active system is not responding, it tries to reconnect starting with the lowest order number.

Troubleshooting Tips for Global Network Management

Troubleshooting Information in the NNMi Help

See the *Troubleshoot Global Network Management* topic in the NNMi help for global network management troubleshooting information.

Clock Synchronization

All NNMi management servers in your network environment that participate in global network management (global managers and regional managers) or single sign-on (SSO) must have their internal time clocks synchronized in universal time. Use a Time Synchronization program, for example, the UNIX (HP-UX / Linux / Solaris) tool Network Time Protocol Daemon (NTPD) or one of the available Windows operating system tools.

If you see the following message at the bottom of the NNMi console:

```
NNMi is not connected to 1 Regional Manager(s). See Help ? System Information, Global Network Management.
```

Check the `nnm.0.0.log` file on the Global Manager for the following message:

```
WARNING: Not connecting to system <serverName> due to clock difference of <number of seconds>. Remote time is <date/time>.
```

Perhaps the clocks have drifted apart and need to be resynchronized. Check the `nnm.0.0.log` file on the Global Manager for the following message:

```
WARNING: Not connecting to system <serverName> due to clock difference of <number of seconds>. Remote time is <date/time>.
```

Within a few minutes of this warning, NNMi disconnects the Regional Manager Connection. And the following message appears at the bottom of the NNMi console:

```
NNMi is not connected to 1 Regional Manager(s). See Help ? System Information, Global Network Management.
```

Global Network Management System Information

Select **Help > System Information**, then click the **Global Network Management** tab to view information about your global network management connections.

Synchronize Regional Manager Discovery from a Global Manager

Suppose you notice an information inconsistency between `global1` and `regional2`. To fix that, run the `nnmnode rediscover.ovpl` script from `global1`, causing `global1` and `regional2` to synchronize. This also results in the `regional2` updating `global1` with any new discovery results.

Consider the network shown in [Figure 8](#) on page 172. Suppose you want `regional2` to synchronize its entire set of nodes, nodes X, Y and Z, with `global1`. Run the following command to synchronize nodes X, Y, and Z with `global1`: `nnmnode rediscover.ovpl -u username -p password -rm regional2`. For more information, see the `nnmnode rediscover.ovpl` reference page, or the UNIX manpage.

Remedying a Destroyed Database on global1

If you take `global1` out of service and need to restore its database, you face several scenarios:

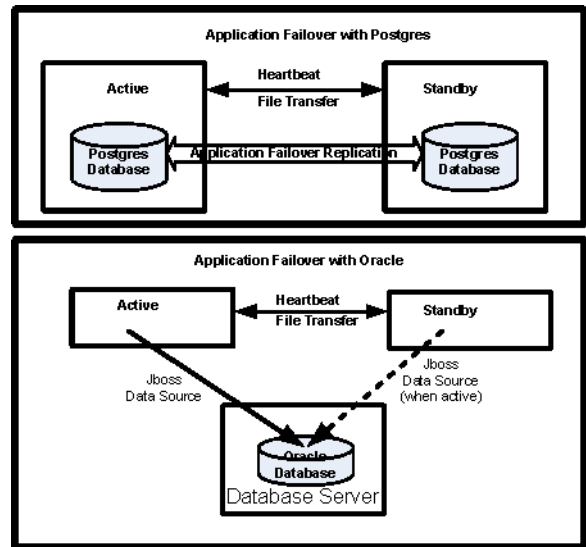
- 1 If you restore `global1`'s database successfully, `regional1` and `regional2` synchronize their cached information with `global1`. There are no manual steps to perform after bringing `global1` back online.
- 2 If `global1` is out of service for an extended period of time, [step 1](#) may not work successfully. To remedy this, run the `nnmnode rediscover.ovpl` script on `global1` to initiate a new discovery on `global1`, `regional1` and `regional2`. In this case you could run status polls on key devices to more quickly get updated status information.

- 3 If you cannot recover `global1`'s database then you need to submit a support call to clear out the old `global1` data from the `regional1` and `regional2` databases using the `nnmsubscription.ovpl` script.

Global Network Management and NNM iSPIs or Third-Party Integrations

Each NNM iSPI or third-party integration has its own unique deployment guideline. For the examples in this chapter, you can deploy some NNM iSPIs on `regional1` only, `global1` only, or on both `regional1` and `global1`. For other NNM iSPIs or third-party integrations, you must have them installed on both `regional1` and `global1`. See the documentation for the NNM iSPI or third-party integration for more information.

Configuring NNMi for Application Failover



Many information technology professionals depend on HP Network Node Manager i Software to notify them when critical network equipment fails and to provide them with a root cause for the failure. They also need NNMi to continue to notify them of network equipment failures, even when the NNMi management server fails. **NNMi application failover** meets this need, transferring application control of NNMi processes from an active NNMi management server to a standby NNMi management server, providing continuance of NNMi functionality.

This chapter contains the following topics:

- [Application Failover Overview](#)
- [Application Failover Basic Setup](#)
- [Configuring NNMi for Application Failover](#)
- [Using the Application Failover Feature](#)
- [iSPIs and Application Failover](#)
- [Integrated Applications](#)
- [Disabling Application Failover](#)
- [Administrative Tasks and Application Failover](#)
- [Network Latency/Bandwidth Considerations](#)

Application Failover Overview

The application failover feature is available for NNMi installations that use either the embedded or Oracle databases. After configuring your systems to use the application failover feature, NNMi detects an NNMi management server failure and triggers a secondary server to assume NNMi functionality.

The following terms and definitions apply to configuring NNMi for application failover:

- **Active:** The server running the NNMi processes.
- **Standby:** The system in the NNMi cluster that is waiting for a failover event; this system is not running NNMi processes.
- **Cluster Member:** A Java process running on a system that is using JGroups technology to connect to a cluster; you can have multiple members on a single system.
- **Postgres:** The embedded database NNMi uses to store information such as topology, incidents, and configuration information.
- **Cluster Manager:** The `nmcluster` process and tool used to monitor and manage the servers for the application failover feature.

Application Failover Basic Setup

To deploy the application failover feature, install NNMi on two servers. This chapter refers to these two NNMi management servers as the **active** and **standby** servers. During normal operation, only the active server is running NNMi services.

The active and standby NNMi management servers are part of a cluster that monitors a heartbeat signal from both of the NNMi management servers. If the active server fails, resulting in the loss of its heartbeat, the standby server becomes the active server.

For application failover to work successfully, the NNMi management servers must meet the following requirements:

- Both NNMi management servers must be running the same type of operating system. For example, if the active server is running an HP-UX operating system, the standby server must also be running an HP-UX operating system.
- Both NNMi management servers must be running the same NNMi version. For example, if NNMi version 9.00 is running on the active server, the identical NNMi version 9.00 must be on the standby server. The NNMi patch levels must also be the same on both servers.
- The system password must be the same on both NNMi management servers.
- For NNMi installations on Windows operating systems, the `%NnmDataDir%` and `%NnmInstallDir%` system variables must be set to identical values on both servers.
- Both NNMi management servers must be running the same database. For example, both NNMi management servers must be running Oracle or both NNMi management servers must be running the embedded database. You cannot mix the two database types if you plan to use the application failover feature.
- Both NNMi management servers must have identical licensing attributes. For example, the node counts and licensed features must be identical.
- Do not enable application failover until NNMi is in an advanced stage of initial discovery. For more information see [Evaluate Discovery](#) on page 66.

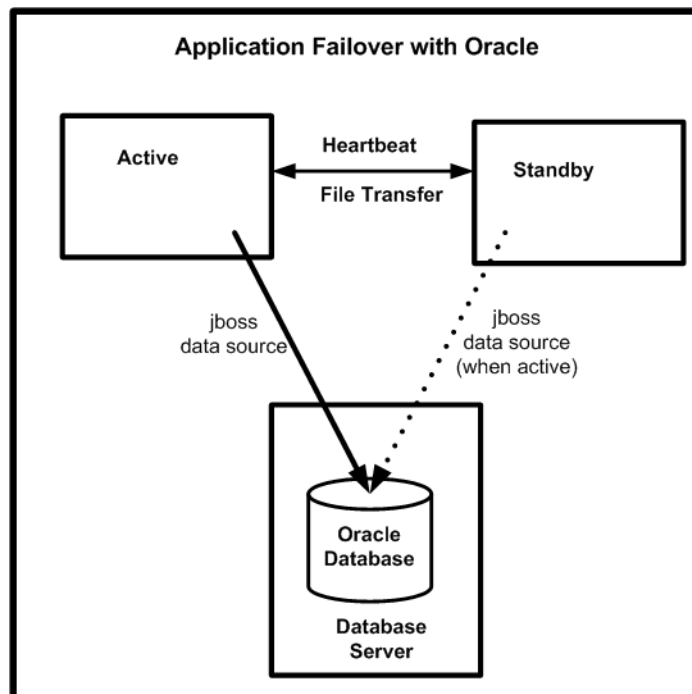
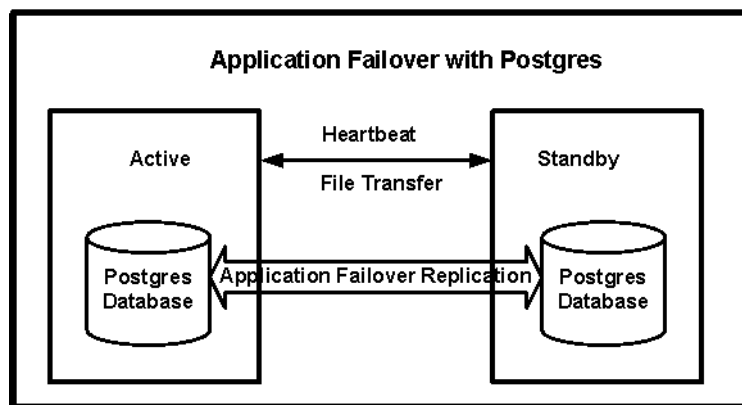
For application failover to function correctly, the active and standby servers must have unrestricted network access to each other. After meeting this condition, complete the steps shown in [Configuring NNMi for Application Failover](#) on page 181. For more information see [NNMi 9.00](#) and [Well-Known Ports](#) on page 547.



Any software that locks files or restricts network access can cause NNMi communication problems. Configure these applications to ignore the files and ports used by NNMi.

Configuring NNMi for Application Failover

- 1 Install NNMi on the active server, server X, and the standby server, server Y, as described in the *NNMi Installation Guide*.



- 2 For each license on server X, obtain a similar non-production license for server Y and install it onto server Y as described in the *License NNMi* section in the *NNMi Installation Guide*.
- 3 Run the `ovstop` command on each server to shut down NNMi.



If you are using application failover with Oracle as your database, your NNMi processes on the standby server should already be stopped.

- 4 Configure server X (active) and server Y (standby) for the application failover feature using guidance from the detailed instructions contained in the `nms-cluster.properties` file. Use the following procedure:



Edit in the following steps means to uncomment the lines in the text block within the file and to modify the text.

- a Edit the following file:

- *Windows*:

```
%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties
```

- *UNIX*: `$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties`

- b Declare a unique name for the NNMi cluster. Use the same name when configuring both the active and standby servers.

```
com.hp.ov.nms.cluster.name=MyCluster
```

- c If the active and standby NNMi management servers reside in different subnets, you must declare two more parameters in the `nms-cluster.properties` file:

```
com.hp.ov.nms.cluster.protocol = TCP
```

```
com.hp.ov.nms.cluster.member.hostnames = fqdn_for_active,  
fqdn_for_standby
```

- d *Optional*. Define other `com.hp.ov.nms.cluster*` parameters within the `nms-cluster.properties` file. Follow the instructions contained within the `nms-cluster.properties` file for modifying each parameter



If you are using application failover with Oracle as your database, NNMi ignores the database parameters contained in the `nms-cluster.properties` file.

- 5 Depending on the approach you take, complete the instructions shown in [Configuring Application Failover to use Self-Signed Certificates](#) on page 103 or the instructions shown in [Configuring Application Failover to use a Certificate Authority](#) on page 104.

- 6 *UNIX*. If server Y is a UNIX server, run the following command:

```
chmod 400 $NnmDataDir/shared/nnm/certificates/nnm.keystore
```

- 7 *UNIX*. If server Y is a UNIX server, run the following command:

```
chmod 400 \  
$NnmDataDir/shared/nnm/conf/nmcluster/cluster.keystore
```

8 Copy the following file from server X to server Y:

- *Windows*:
%NnmDataDir%\shared\nnm\conf\nnmcluster\cluster.keystore
- *UNIX*:
\$NnmDataDir/shared/nnm/conf/nnmcluster/cluster.keystore

9 On server X, start the NNMi cluster manager:

```
nnmcluster -daemon
```

After you run the **nnmcluster -daemon** command on NNMi management server X, the NNMi cluster manager goes through the following startup routine:

- Connects NNMi management server X to the cluster.
- Detects that there are no other NNMi management servers present.
- NNMi management server X assumes the active state.
- Starts the NNMi services on NNMi management server X (the active server).
- Creates a database backup.

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

10 Wait a few minutes for server X to become the first active node in the cluster. Run the **nnmcluster -display** command on server X and search the displayed results for the term ACTIVE as in ACTIVE_NNM_STARTING or ACTIVE_SomeOtherState. Do not continue with [step 11](#) until you know that server X is the active node.

11 On server Y, start the NNMi cluster manager:

```
nnmcluster -daemon
```

After you run the **nnmcluster -daemon** command on NNMi management server Y, the NNMi cluster manager goes through the following startup routine:

- Connects NNMi management server Y to the cluster.
- Detects that NNMi management server X is present and is in the active state. The display shows STANDBY_INITIALIZING.
- Compares the database backup on NNMi management server Y to the backup on NNMi management server X. If these do not match, a new database backup is sent from NNMi management server X (active) to NNMi management server Y (standby). The display shows STANDBY_RECV_DBZIP.
- NNMi management server Y receives a minimal set of transaction logs which is the minimum necessary for the backup to be applicable for its standby state. The display shows STANDBY_RECV_TXLOGS.
- NNMi management server Y goes into a waiting state, continuously receiving new transaction logs and heartbeat signals from NNMi management server X. The display shows STANDBY_READY.

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

- 12 If a failover occurs, the NNMi console for server X no longer functions. Close the NNMi console session for server X and sign in on server Y (the new active server). Instruct NNMi users to store two bookmarks in their browsers, one to server X (the active NNMi management server) and one to server Y (the standby NNMi management server). If a failover occurs, users can connect to server Y (the standby NNMi management server).
- 13 Instruct network operations center (NOC) personnel to configure their devices to send traps to both server X and server Y. While server X (active) is running, it processes the forwarded traps and server Y (standby) ignores the forwarded traps.

Using the Application Failover Feature

Now that you have both NNMi management servers running the cluster manager, with one active node and one standby node, you can use the cluster manager to view the cluster status. The cluster manager has three modes:

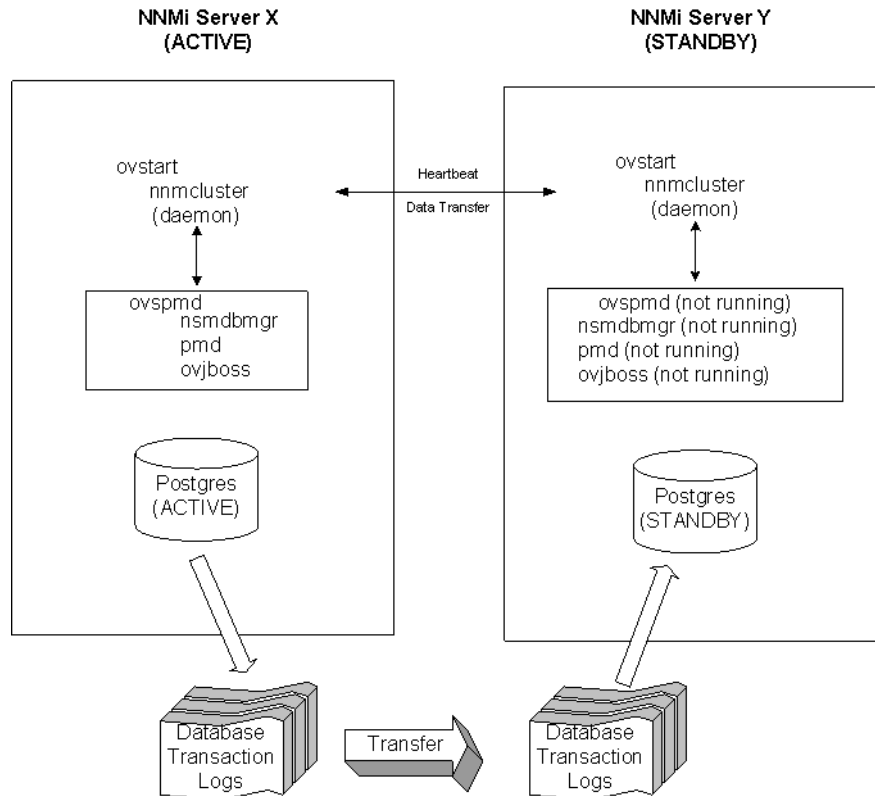
- **daemon mode:** The cluster manager process runs in the background, and uses the `ovstop` and `ovstart` commands to start and stop the NNMi services.
- **interactive mode:** The cluster manager runs an interactive session in which the NNMi administrator can view and change cluster attributes. For example, the NNMi administrator can use this session to enable or disable the application failover feature or shut down the daemon processes.
- **command line mode:** The NNMi administrator views and changes cluster attributes at the command prompt.

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

Application Failover Behavior Using the Embedded Database

Figure 9 shows the application failover configuration for two NNMi management servers using the embedded database. Refer to this figure while reading the rest of this chapter.

Figure 9 Application Failover Configuration (embedded database)



After you start both the active and standby nodes, the standby node detects the active node, requests a database backup from the active node, but does not start NNMi services. This database backup is stored as a single Java-ZIP file. If the standby node already has a ZIP file from a previous cluster-connection, and NNMi finds that the file is already synchronized with the active server, the file is not retransmitted.

While both the active and standby nodes are running, the active node periodically sends database transaction logs to the standby node. You can modify the frequency of this data transfer by changing the value of the `com.hp.ov.nms.cluster.timeout.archive` parameter in the `nms-cluster.properties` file. These transaction logs accumulate on the standby node, and are available on the standby node any time it needs to become active.

When the standby node receives a full database backup from the primary node, it places the information into its embedded database. It also creates a `recovery.conf` file to inform the embedded database that it should consume all received transaction logs before it becomes available to other services.

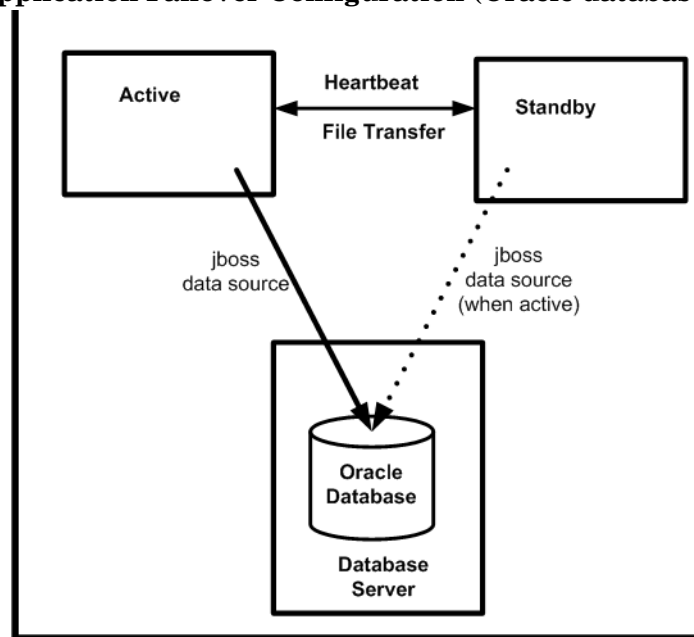
If the active node becomes unavailable for any reason, the standby node becomes active by running an `ovstart` command to start the NNMi services. The standby NNMi management server imports the transaction logs before starting the remaining NNMi services.

If the active NNMi system fails, the standby system begins discovery and polling activities. This transition keeps NNMi monitoring and polling your network while you diagnose and repair the failed system.

Application Failover Behavior Using an Oracle Database

Figure 10 shows the application failover configuration for two NNMi management servers using an Oracle database. Refer to this figure while reading the rest of this chapter.

Figure 10 Application Failover Configuration (Oracle database)



If the active node becomes unavailable for any reason, the standby node becomes active by running an `ovstart` command to start the NNMi services.

If the active NNMi system fails, the standby system begins discovery and polling activities. This transition keeps NNMi monitoring and polling your network while you diagnose and repair the failed system.

Application Failover Scenarios

There are several possible problems that can cause the active NNMi management server to stop sending heartbeats, and to initiate a failover:

- Scenario 1: The active NNMi management server fails.
- Scenario 2: The system administrator shuts down or reboots the active NNMi management server.
- Scenario 3: The NNMi administrator shuts down the cluster.
- Scenario 4: The network connection between the active and the standby NNMi management servers fails.

In scenario 4, both NNMi management servers run in the active state. When the network device comes back online, the two NNMi management servers automatically negotiate which node should become the new active node.

Additional ovstart and ovstop Options

When you use the **ovstop** and **ovstart** commands on NNMi management servers configured for application failover, NNMi actually runs the following commands:

- **ovstart:** `nnmcluster -daemon`
- **ovstop:** `nnmcluster -disable -shutdown`



If you run an **ovstop** command, NNMi does not failover to the standby node. HP designed the **ovstop** command to support temporary maintenance stoppages. To manually initiate a failover, use the **-failover** option with the **ovstop** command. For more information, see the *ovstop* reference page, or the UNIX manpage.

The following options to the **ovstop** command apply to NNMi management servers configured in an application failover cluster:

- **ovstop -failover:** This command stops the local daemon-mode cluster process and forces a failover to the standby NNMi management server. If the failover mode was previously disabled, it is re-enabled. This command is equivalent to: `nnmcluster -enable -shutdown`
- **ovstop -nofailover:** This command disables failover mode and then stops the local daemon-mode cluster process. No failover occurs. This command is equivalent to: `nnmcluster -disable -shutdown`
- **ovstop -cluster:** This command stops both the active and standby nodes, removing them both from the cluster. This command is equivalent to: `nnmcluster -halt`

Application Failover Incidents

Any time the `nnmcluster` process or someone using the `nnmcluster` command starts a node as active, NNMi generates one of the following incidents:

- *NnmClusterStartup:* The NNMi cluster was started, and no active node was present. Therefore the node was started in the active state. This incident has a NORMAL severity.
- *NnmClusterFailover:* The NNMi cluster detected a failure of the active node. The standby node was then enabled and NNMi services started on the new active node. This incident has a MAJOR severity.

iSPIs and Application Failover

You can use the application failover feature for a Smart Plug-in (iSPI) that you deploy along with NNMi if the deployment meets the following requirements:

- The NNM iSPI runs on the NNMi management server.
- The NNM iSPI uses the same embedded database instance as NNMi.

The NNM iSPI for QA, the NNM iSPI for Metrics, and the NNM iSPI for Traffic are exceptions to this description. If you plan to configure the NNMi application failover feature, you must install these iSPIs on dedicated servers. In this case, the iSPIs automatically connect to the new NNMi management server after failover occurs. As

part of NNMi application failover configuration, run the enablement script for the NNM iSPI for Metrics, the NNM iSPI for QA, or the NNM iSPI for Traffic on each NNMi management server in the cluster.

For more information, see *Support for Application Failover* in the NNM iSPI for Metrics, the NNM iSPI for QA, or the NNM iSPI for Traffic help.

NNM iSPI Installation Information

To install an NNM iSPI on an NNMi management server that is already part of an application failover cluster, do the following:

- 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](#) on page 38.
- 2 As a precaution, back up the NNMi data on both the active and standby NNMi management servers before proceeding. For information, see [Backup Scope](#) on page 257.
- 3 Embedded database only: As a precaution, on the active NNMi management server run the **nnmcluster -syncdb** command and wait for the command to complete.
- 4 Copy the `nnm.keystore` and `nnm.truststore` files from Server Y to some temporary location on Server X. The remaining steps refer to these file locations as `<keystore>` and `<truststore>`.
- 5 Run the following script on Server X to merge Server Y's certificates into Server X's `nnm.keystore` and `nnm.truststore` files.

Windows:

```
nnmcertmerge.ovpl -keystore <keystore> -truststore <truststore>
```

UNIX:

```
nnmcertmerge.ovpl -keystore <keystore> -truststore <truststore>
```

- 6 Copy the merged `nnm.keystore` and `nnm.truststore` files from server X to server Y, so that both nodes have the merged files. The location of these files is as follows:
 - *Windows:* %NNM_DATA%\shared\nnm\certificates
 - *UNIX:* \$NNM_DATA/shared/nnm/certificates
- 7 On the standby NNMi management server, run the following command:

```
nnmcluster -shutdown
```
- 8 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
- 9 Comment out the `com.hp.ov.nms.cluster.name` option and save the file.
- 10 Run the **ovstart** command on the standby NNMi management server. This brings up NNMi services in the standalone (unclustered) state.
- 11 Install the NNM iSPI on the standby NNMi management server as described in the iSPI installation guide.

- 12 Run the **nnmcluster -halt** command on the active NNMi management server.
- 13 Edit the following file on the active NNMi management server:
 - *Windows*:
%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties
 - *UNIX*: \$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties
- 14 Comment out the **com.hp.ov.nms.cluster.name** option and save the file.
- 15 Run the **ovstart** command on the active NNMi management server. This brings up NNMi services in the standalone (unclustered) state.
- 16 Install the NNM iSPI on the active NNMi management server as described in the iSPI installation guide.
- 17 Edit the following file on **both** the active and standby NNMi management servers:
 - *Windows*:
%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties
 - *UNIX*: \$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties
- 18 Uncomment the **com.hp.ov.nms.cluster.name** option and save each file.
- 19 Run the **ovstart** command on the active NNMi management server.
- 20 Wait a few minutes for the active NNMi management server to become the first active node in the cluster. Run the **nnmcluster -display** command on the active NNMi management server and search the displayed results for the term **ACTIVE** as in **ACTIVE_NNM_STARTING** or **ACTIVE_SomeOtherState**. Do not continue with [step 21](#) until you know that the active NNMi management server is the active node.
- 21 Run the **ovstart** command on the standby NNMi management server.

Integrated Applications

When other HP Software or third-party products are integrated with NNMi, the affect of NNMi application failover on an integration depends on how a product communicates with NNMi. For more information, see the appropriate chapter in the [Integrations with NNMi](#) section.

If an integrated product must be configured with information about the NNMi management server, the following information applies:

- If long-term, you can update the NNMi management server information within the integrating product configuration. For more information, see the appropriate chapter in the [Integrations with NNMi](#) section.
- If the outage appears to be temporary, you can resume using the integrating product after server X returns to service. To return server X to service, follow these steps:

- 1 On server X, run the following command:

```
nnmcluster -daemon
```

Server X joins the cluster and assumes a standby state.

- 2 On server X, run the following command:

```
nnmcluster -acquire
```

Server X changes to the active state.

If you anticipate that the original server X will be out of service for a longer time, you can update the NNMi management server IP address within the integrating product. For instructions on how to modify the IP address field, see the integrating product documentation.

Disabling Application Failover

Suppose you configure application failover, use it for a few days, then decide to completely disable it. The following information explains how to completely disable application failover. Complete the following instructions, including actions on both the active and standby NNMi management servers configured in the application failover cluster.

- 1 Run **nnmcluster -enable** command on the *active* NNMi management server.
- 2 Run the **nnmcluster -shutdown** command on the *active* NNMi management server.
- 3 Wait a few minutes for the old standby NNMi management server to become the new active NNMi management server.
- 4 Run the **nnmcluster -display** command on the new active (old standby) NNMi management server.
- 5 Search the displayed results for the ACTIVE_NNM_RUNNING status. Repeat [step 4](#) until you see the ACTIVE_NNM_RUNNING status.
- 6 Run the **nnmcluster -shutdown** command on the new active (old standby) NNMi management server.
- 7 Run the **nnmcluster -display** command repeatedly on the new active (old standby) until you no longer see a DAEMON process.
- 8 Edit the following file both NNMi management servers configured in the cluster:
 - *Windows*:
%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties
 - *UNIX*: \$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties
- 9 Comment out the com.hp.ov.nms.cluster.name option on both NNMi management servers and save each file.
- 10 Edit the following file on both NNMi management servers:
 - *Windows*:
%NnmDataDir%\shared\nnm\databases\Postgres\postgresql.conf
 - *UNIX*: \$NnmDataDir/shared/nnm/databases/Postgres/postgresql.conf
- 11 Remove the following lines that begin with `archive_command` and `archive_timeout` in each file. This is an example of what these lines could look like on a Windows NNMi management server. These lines might look slightly different on your server.

```
archive_command = 'nmcluster.exe -archive "%p" "C:/Documents  
and Settings/All Users/Application Data/HP/HP BTO Software/  
shared/nnm/databases/Postgres_standby/TxWALs_send"/%f'  
  
archive_timeout = 900
```

Make sure to save your changes.

- 12 If these are Windows NNMi management servers, navigate to the Services (Local) console and do the following on each server:
 - a Set the Startup type for the HP NNM Cluster Manager to Disabled.
 - b Set the Startup type for the HP Openview Process Manager to Automatic.
- 13 Run the **ovstart** command on the former active NNMi management server only. In the application failover configuration, this is the NNMi management server that has a permanent NNMi license.
- 14 If you were using a secondary node license on the former standby server. Do not run the **ovstart** command on the former standby NNMi management server. In the application failover configuration, this is the NNMi management server that has a secondary node license. To run this NNMi management server as a standalone server, you must purchase and install a permanent license. See the NNMi Installation Guide for more information.
- 15 If both NNMi management servers start successfully, then remove the following directory from both the standby and active NNMi management servers:
 - *Windows:* %NnmDataDir%\shared\nnm\databases\Postgres_standby
 - *UNIX:* \$NnmDataDir/shared/nnm/databases/Postgres_standby



This directory is a default directory and is the value of the `com.hp.ov.nms.cluster.archivedir` parameter located in the `nms-cluster.properties` file. These instructions assume you did not change this value. If you changed the value of the `com.hp.ov.nms.cluster.archivedir` parameter in the `nms-cluster.properties` file, then remove the directory that equates to the new value.

- 16 Remove the following directory from both the standby and active NNMi management servers:
 - *Windows:* %NnmDataDir%\shared\nnm\databases\Postgres.OLD
 - *UNIX:* \$NnmDataDir/shared/nnm/databases/Postgres.OLD

Administrative Tasks and Application Failover

The following information explains how to effectively manage application failover when doing administrative tasks such as patching and restarting NNMi management servers.

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](#) on page 38.
- 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](#) on page 257.
- 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 `syncdb`, 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 193.
 - 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 `nmcluster -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 for QA, the NNM iSPI for Metrics, or the NNM iSPI 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 `nmconfigexport.ovpl` script on both the active and standby NNMi management servers before proceeding. For information, see [Best Practice: Save the Existing Configuration](#) on page 38.
- 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](#) on page 257.

- 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 **syncdb**, 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 Apply the NNMi patch to the active NNMi management server using the instructions provided with the patch.
- 7 Run the **ovstart** command on the active NNMi management server.
- 8 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.
- 9 Run the **nnmcluster -syncdb** command to create a new backup.
- 10 Apply the NNMi patch to the standby NNMi management server.
- 11 Run the **ovstart** command on the standby NNMi management server.
- 12 If you installed the NNM iSPI for QA, the NNM iSPI for Metrics, or the NNM iSPI 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.
- 13 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 Restarting the NNMi Management Servers

You can restart the standby NNMi management server at any time with no special instructions. If you restart both the standby and active NNMi management servers, restart the primary NNMi management server first.

To restart either the active or the standby NNMi management server, do the following.

- 1 Run the `nnmcluster -disable` command on the NNMi management server to disable the application failover feature.
- 2 Restart the NNMi management server.
 - a Run the `ovstop` command on the NNMi management server.
 - b Run the `ovstart` command on the NNMi management server.
- 3 Run the `nnmcluster -enable` command on the NNMi management server to enable the application failover feature.

Application Failover and Recovery from a Previous Database Backup (Embedded Database Only)

To restore your NNMi database from an original backup when active and standby NNMi management servers are configured for application failover, follow these steps:

- 1 Run the `nnmcluster -halt` command on the active NNMi management server.
- 2 Delete or move the following directory on both the active and standby NNMi management servers:
 - *Windows:* %NnmDataDir%\shared\nnm\databases\Postgres_standby
 - *UNIX:* \$NnmDataDir/shared/nnm/databases/Postgres_standby
- 3 Restore the database on the primary NNMi management server:
 - a Modify the following file to comment out the cluster name:
 - *Windows:*
%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties
 - *UNIX:* \$NnmDataDir/shared/nnm/conf/props\nms-cluster.properties
 - b Restore the database as normal. See [Restoring NNMi Data](#) on page 259.
 - c Run the `ovstop` command on the active NNMi management server.
 - d Modify the following file to uncomment the cluster name:
 - *Windows:*
%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties
 - *UNIX:* \$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties
- 4 Run the `ovstart` command on the active NNMi management server.
- 5 Wait until the active NNMi management server generates a new backup. To verify that this step is complete, run the `nnmcluster -display` command and look for an `ACTIVE_NNM_RUNNING` message.
- 6 Run the `ovstart` command on the standby NNMi management server. The standby NNMi management server copies and extracts the new backup. To verify that this step is complete, run the `nnmcluster -display` command and look for a `STANDBY_READY` message.

Network Latency/Bandwidth Considerations

NNMi application failover works by exchanging a continuous heartbeat signal between the nodes in the cluster. It uses this same network channel for exchanging other data files such as the NNMi embedded database, database transaction logs, and other NNMi configuration files. HP recommends using a high performance, low latency connection for NNMi application failover when implementing it over a WAN (wide area network).

The NNMi embedded database can become quite large, and can grow to 1GB or more even though this file is always compressed. Also, NNMi generates hundreds, or even thousands, of transaction logs during the built-in backup interval (a configuration parameter that defaults to 24 hours). Each transaction log can be several megabytes, up to a maximum size of 16MB. (These files are also compressed). Example data collected from an HP test environments is shown here:

Number of nodes managed: 15,000

Number of interfaces: 100,000

Time to complete spiral discovery of all expected nodes: 12 hours

Size of database: 850MB (compressed)

During initial discovery: ~10 transaction logs per minute (peak of ~15/min)

10 TxLogs/minute X 12 hours = 7200 TxLogs @ ~10MB = ~72GB

This is a lot of data to send over the network. If the network between the two nodes is unable to keep up with the bandwidth demands of NNMi application failover, the standby node can fall behind in receiving these database files. This could result in a larger window of potential data loss if the active server fails.

Similarly, if the network between the two nodes has a high latency or poor reliability, this could result in a *false* loss-of-heartbeat between the nodes. For example, this can happen when the heartbeat signal does not respond in a timely manner, and the standby node assumes that the active node has failed. There are several factors involved in detecting loss-of-heartbeat. NNMi avoids false failover notification as long as the network keeps up with the application failover data transfer needs.

In HP's verification of multi-subnet NNMi application failover, the active and standby servers resided in the United States, one in Colorado and another in Virginia. This provided acceptable bandwidth and latency, with no false failovers.

Application Failover and the NNMi Embedded Database

Application failover works with both the embedded and the Oracle database for NNMi 9.00. However, with Oracle, the database resides on a server that is separate from any NNMi management server. When you configure NNMi to work with an Oracle database, there is no database replication. This results in reduced network demands for application failover using an Oracle database. When using application failover with Oracle, the network uses less than 1% of the network demands as compared to using application failover with the embedded database. The information contained in this section explains NNMi traffic information related to application failover using the embedded database.

After you configure NNMi using the embedded database for application failover, NNMi does the following:

- 1 The active node performs a database backup, storing the data in a single ZIP file.
- 2 NNMi sends this ZIP file across the network to the standby node.
- 3 The standby node expands the ZIP file, and configures the embedded database to import transaction logs on the first startup.
- 4 The embedded database on the active node generates transaction logs, depending on database activity.
- 5 Application failover sends the transaction logs across the network to the standby node, where they accumulate on the disk.
- 6 When the standby node becomes active, NNMi starts, and the database imports all transaction logs across the network. The amount of time this takes depends on the number of files and complexity of the information stored within those files (some files take longer to import than other files of comparable size).
- 7 After the standby node imports all of the transaction logs, the database becomes available, and the standby node starts the remaining NNMi processes.
- 8 The original standby node is now active, and the procedure starts over at [step 1](#).

Network Traffic in and Application Failover Environment

NNMi transfers many items across the network from the active node to the standby node in an application failover environment:

- Database Activity: the database backup, as a single ZIP file.
- Transaction logs.
- A periodic *heartbeat* so that each application failover node verifies that the other node is still running.
- File comparison lists so that the standby node can verify that its files are in sync with those on the active node.
- Miscellaneous events, such as changes in parameters (enable/disable failover and others) and nodes joining or node leaving the cluster,

The first two items generate 99% of the network traffic used by application failover. This section explores these two items in more detail.

Database Activity: NNMi generates transaction logs for all database activity. Database activity includes everything in NNMi. This activity includes, but is not limited to, the following database activities:

- Discovering new nodes.
- Discovering attributes about nodes, interfaces, VLANs, and other managed objects.
- State polling and status changes.
- Incidents, events, and root cause analysis.
- Operator actions in the NNMi console.

Database activity is outside of your control. For example, an outage on the network results in NNMi generating many incidents and events. These incidents and events trigger state polling of devices on the network, resulting in updates to device status in NNMi. When the outage is restored, additional *node up* incidents result in further status changes. All of this activity updates entries in the database.

Although the embedded database itself grows with database activity, it reaches a stable size for your environment, with only moderate growth over time.

Database Transaction Logs: The embedded database works by creating an empty 16 MB file, then writing database transaction information to that file. NNMi closes this file, then makes it available to application failover after 15 minutes, or after writing 16 MB of data to the file, whichever comes first. That means that a completely idle database will generate one transaction log file every 15 minutes, and this file will be essentially *empty*. Application failover compresses all transaction logs, so an empty 16 MB file compresses down to under 1MB. A *full* 16MB file compresses to about 8 MB. Keep in mind that during periods of higher database activity, application failover generates more transaction logs in a shorter period of time, since each file gets full faster.

An Application Failover Traffic Test

The following test resulted in an average of about 2 transaction log files per minute, with an average file size of 7 MB per file. This is due to the database activity associated with discovery of the additional 5000 nodes added with each failover event. The database in this test case eventually stabilized at about 1.1GB (as measured by the size of the backup ZIP file), with 31,000 nodes and 960,000 interfaces.

Testing Method: During the first 4 hours, test personnel seeded NNMi with 5,000 nodes and waited until discovery stabilized. After 4 hours, test personnel induced failover (the standby node became active, and the previous-active node became standby). Immediately after failover, test personnel added approximately 5,000 more nodes, waited another 4 hours to let the NNMi discovery process stabilize, then induced another failover (failed back to the previous active node). Test personnel repeated this cycle several times with some variation in the time between failover (4 hours, then 6 hours, then 2 hours). After each failover event, test personnel measure the following:

- The size of the database backup ZIP file (created when the node first became active).
- The transaction logs: the total number of files and disk space utilization.
- The number of nodes and interfaces in the NNMi database immediately before inducing failover.
- Time to complete failover. This included the time from the initial `ovstop` command on the active node until the standby node became fully active with NNMi running.

Table 13 summarizes the results:

Table 13 Application Failover Test Results

Hours	DB.zip Size (MB)	No. of Tx Logs	Tx Logs (GB)	Nodes	Interfaces	Failover Time (Minutes)
4	6.5	50	.3	5,000	15,000	5
8	34	500	2.5	12,000	222,000	10
12	243	500	2.5	17,000	370,000	25
16	400	500	3.5	21,500	477,000	23
20	498	500	3.5	25,500	588,000	32
26	618	1100	7.5	30,600	776,000	30
28	840	400	2.2	30,600	791,000	31
30	887	500	2.5	30,700	800,000	16

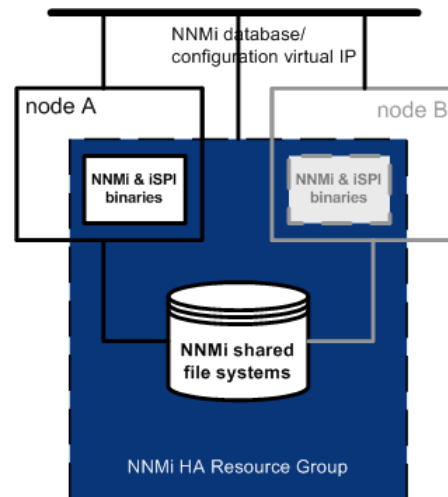
Observations: When NNMi transferred files from the active node to the standby node, the transfer averaged about 5 GB every 4 hours, which is a continuous throughput of approximately 350KB/s (kilobytes per second) or 2.8 Mb/S (megabits per second).

- ▶ This data does not include any other application failover traffic, such as the heartbeat, file consistency checks, or other application failover communication. This data also excludes the overhead of network I/O, such as packet headers. This data only included the actual network payload of each file's contents moving across the network.
- ▶ The traffic generated by NNMi application failover environment is very bursty. Application failover identifies new transaction logs on the active node every five minutes and sends these logs to the standby node. Depending on network speed, the standby node should receive all of the new files in a short time, resulting in a relatively idle network for the remainder of that 5-minute interval.

Every time the active and standby nodes switch roles (the standby node becomes active and the active node becomes standby), the new active node will generate a complete database backup and send this across the network to the new standby node. This database backup also occurs periodically, backing up every 24 hours by default. Every time NNMi generates a new backup, it sends this backup to the standby node. Having this new backup available on the standby node reduces the failover time, as all of the transaction logs NNMi generated in that 24 hour interval are already in the database, and do not need to be imported at failover time.

The information provided in the above section will help you understand how the network might perform after a failover when using NNMi with application failover using the embedded database.

Configuring NNMi in a High Availability Cluster



High availability (HA) refers to a hardware and software configuration that provides for uninterrupted service should some aspect of the running configuration fail. An HA cluster defines a grouping of hardware and software that works together to ensure continuity in functionality and data when failover occurs.

NNMi provides support for configuring NNMi to run in an HA cluster under one of several separately purchased HA products. Most of the NNM Smart Plug-ins (iSPIs), but not the NNM iSPI NET diagnostics server, can also run under HA. (The NNM iSPI for Traffic does not currently run under HA.)

This chapter provides a template for configuring NNMi to run in an HA environment. This chapter does not provide end-to-end instructions for configuring your HA product. The HA configuration commands that NNMi provides are wrappers around the commands for the supported HA products. If you prefer, you can substitute the HA product-specific commands where these instructions specify NNMi-provided commands.

If you plan to install any NNM iSPIs on the NNMi management server, also see the deployment guides for those NNM iSPIs.

This chapter contains the following topics:

- [Supported HA Products](#) on page 202
- [Prerequisites to Configuring NNMi for HA](#) on page 202
- [HA Concepts](#) on page 203
- [Configuring HA](#) on page 209
- [Shared NNMi Data](#) on page 218
- [Licensing NNMi in an HA Cluster](#) on page 220
- [Maintaining the HA Configuration](#) on page 221
- [Unconfiguring HA](#) on page 225
- [Patching NNMi under HA](#) on page 229
- [Upgrading NNMi under HA from NNMi 8.1x to NNMi 9.00](#) on page 230
- [Troubleshooting the HA Configuration](#) on page 234
- [HA Configuration Reference](#) on page 239

Supported HA Products

The NNMi-provided commands for configuring and running NNMi under HA work with the following HA products for the designated operating systems:

- *HP-UX*: HP Serviceguard version 11.18 or higher
- *Linux*:
 - Veritas Cluster Server (VCS) version 5.0
 - HP Serviceguard version 11.18
- *Solaris*: Veritas Cluster Server (VCS) version 5.0
- *Windows*:
 - Microsoft Failover Clustering for Windows Server 2008 (MSFC)
 - Microsoft Cluster Service for Windows Server 2003 (MSCS)

While you can follow the template in this chapter to configure NNMi to run under other HA products, HP does not provide support for cluster configuration issues for other configurations.

Prerequisites to Configuring NNMi for HA

Any system that you want to include as a node in an NNMi HA cluster must meet the following requirements:

- Supports the use of a virtual IP address.
- Supports the use of a shared disk.
- Meets all requirements for NNMi as described in the *HP NNMi Software System and Device Support Matrix*.
- Includes the following additional patches:
 - *Windows*:
 - Microsoft hotfix for *Connecting to SMB share on a Windows 2000-based computer or a Windows Server 2003-based computer may not work with an alias name*, which is available from **<http://support.microsoft.com/?id=281308>**
 - *UNIX*: No known additional requirements.
- Meets all requirements described in the documentation for the HA product on which you plan to run NNMi.

Before you begin to configure NNMi for HA, use the commands for your HA product to configure and test an HA cluster. The HA cluster provides such functionality as checking the application heartbeat and initiating failover. The HA cluster configuration must, at a minimum, include the following items:

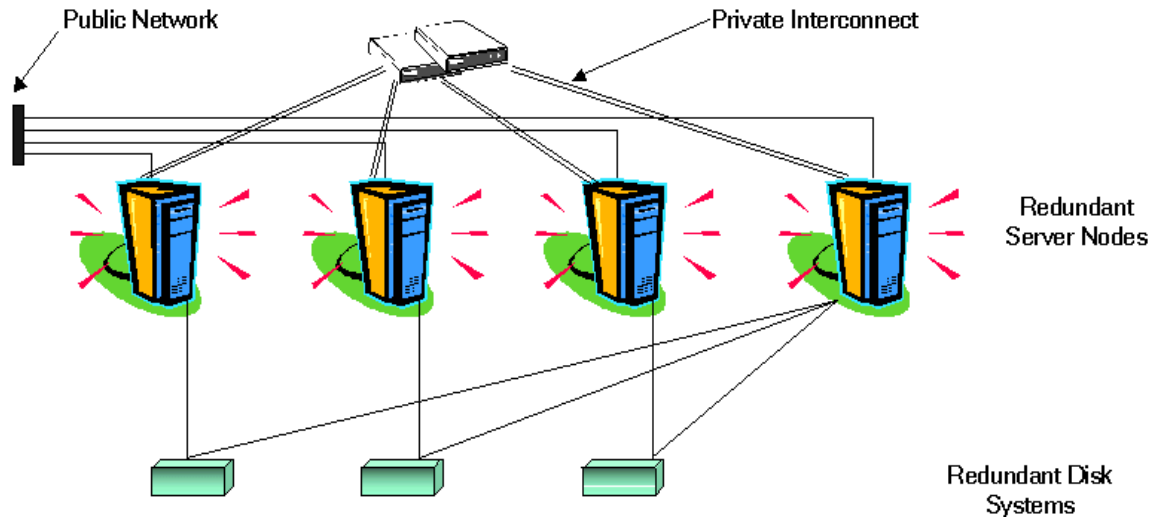
- (UNIX only) ssh
- (UNIX only) remsh
- Virtual IP address for the HA cluster that is DNS-resolvable

- Virtual hostname for the HA cluster that is DNS-resolvable

HA Concepts

Cluster architecture provides a single, globally coherent process and resource management view for the multiple nodes of a cluster. [Figure 11](#) shows an example of a cluster architecture.

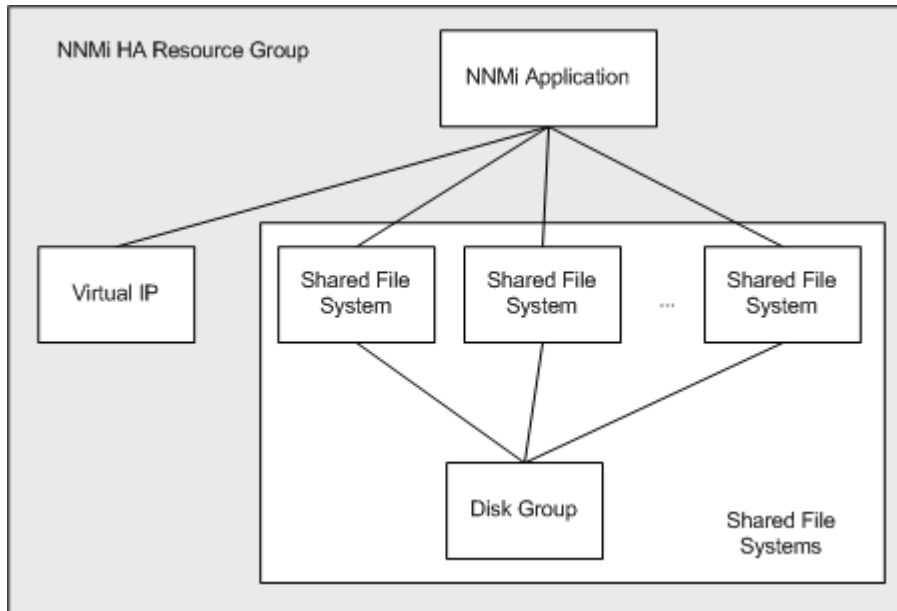
Figure 11 Architecture of a High Availability Cluster



Each node in a cluster connects to one or more public networks and also connects to a private interconnect, representing a communication channel for transmitting data between cluster nodes.

In modern cluster environments such as HP Serviceguard, Veritas Cluster Server, Microsoft Failover Clustering, or Microsoft Cluster Services, applications are represented as compounds of resources, which are simple operations that enable applications to run in a cluster environment. The resources construct an **HA resource group**, which represents an application running in a cluster environment. [Figure 12](#) shows an example HA resource group.

Figure 12 Typical HA Resource Group Layout



This document uses the term *HA resource group* to designate a set of resources in any cluster environment. Each HA product uses a different name for the HA resource group. [Table 14](#) lists the term for each supported HA product that equates to *HA resource group* for this document. (For the specific supported versions of each HA product, see [Supported HA Products](#) on page 202.)

Table 14 Terminology for HA Resource Group in the Supported HA Products

HA Product	Abbreviation	Equivalent Term for HA Resource Group
HP Serviceguard	SG	Package
Veritas Cluster Server	VCS	Service Group
Microsoft Failover Clustering	MSFC	Resource Group
Microsoft Cluster Services	MSCS	Resource Group

HA Terms

[Table 15](#) lists and defines some common HA terms.

Table 15 Common HA Terms

Term	Description
HA resource group	An application running in a cluster environment (under an HA product). An HA resource group can simultaneously be a cluster object that represents an application in a cluster.
Volume group	One or more disk drives that are configured to form a single large storage area.
Logical volume	An arbitrary-size space in a volume group that can be used as a separate file system or as a device swap space.

Table 15 Common HA Terms (cont'd)

Term	Description
Primary cluster node	The first system on which the software product is installed, <i>and</i> the first system on which HA is configured. The shared disk is mounted on the primary cluster node for initial set up. The primary cluster node generally becomes the first active cluster node, but you do not need to maintain the primary designation after HA configuration is complete. The next time you update the HA configuration, another node might become the primary cluster node.
Secondary cluster node	Any system that is added to the HA configuration after the primary cluster node has been fully configured for HA.
Active cluster node	The system that is currently running the HA resource group.
Passive cluster node	Any system that is configured for HA but is not currently running the HA resource group. If the active cluster node fails, the HA resource group fails over to one of the available passive cluster nodes, which then becomes the active cluster node for that HA resource group.

NNMi HA Cluster Scenarios

For NNMi HA configuration, NNMi is installed on each system that will become part of an HA resource group. The NNMi database is located on a separate disk that is accessed by the NNMi programs running on each system. (Only one system, the active cluster node, accesses the shared disk at any given time.)

This approach is valid for the embedded and third-party database solutions.



Run the NNMi database backup and restore scripts on the active cluster node only.

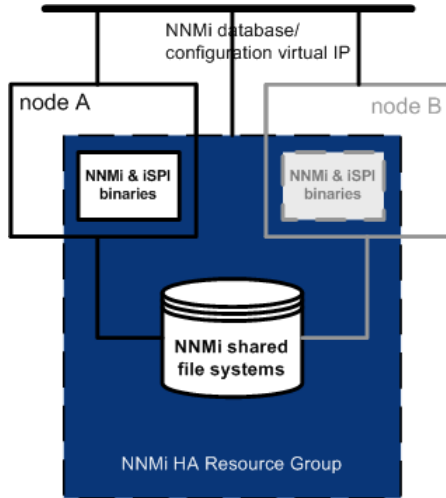
NNMi-only scenario

Figure 13 shows a graphical representation of the NNMi HA cluster scenario. In this figure the NNMi HA resource group is synonymous with the NNMi HA cluster.

Node A and node B are each a fully installed NNMi management server that contains the NNMi program and any NNM iSPIs that run on that system. The active cluster node accesses the shared disk for runtime data. Other products connect to NNMi by means of the virtual IP address of the HA resource group.

If the cluster contains more than two NNMi nodes, additional nodes are configured similarly to node B in Figure 13.

Figure 13 Basic Scenario for NNMi HA Cluster

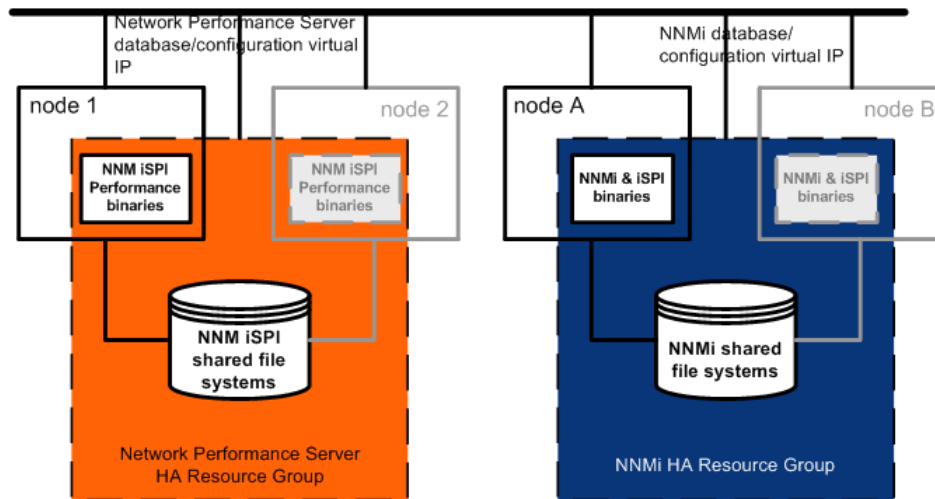


For information about how to implement this scenario, see [Configuring NNMi for HA](#) on page 209 and [Configuring NNM iSPIs for HA](#) on page 215.

**NNMi and
NNM iSPI
Performance products
on a dedicated server
scenario**

If you are running any of the NNM iSPI Performance products on a dedicated server, you can configure these NNM iSPIs to run as a separate HA resource group within the NNMi HA cluster, as shown in [Figure 14](#). The NNMi HA resource group is the same as that described for the NNMi-only scenario.

Figure 14 HA for NNMi and NNM iSPI Performance Products on a Dedicated Server



For information about how to implement this scenario, see [Configuring NNMi for HA](#) on page 209 and [Configuring NNM iSPIs for HA](#) on page 215.

Other options for the NNM iSPI Performance products on a dedicated server are as follows:

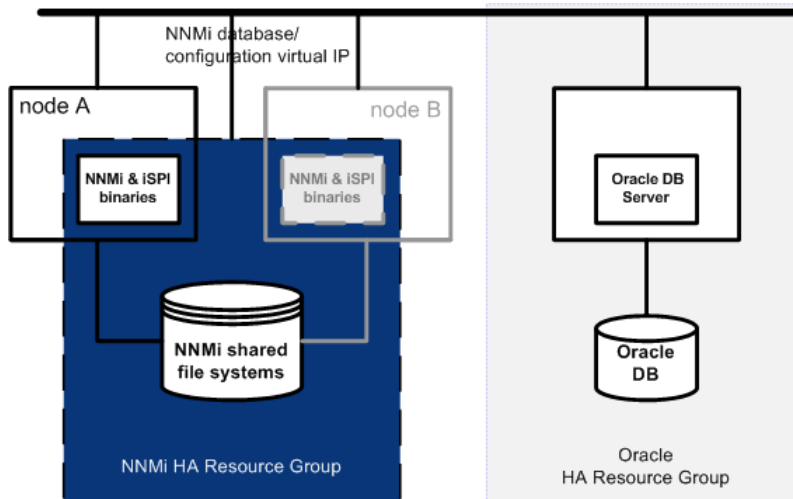
- Run the NNM iSPI Performance products on a single system with no HA. Use this approach while evaluating the NNM iSPIs and for environments where it is not critical for performance data to be always available.
- Configure the NNM iSPI Performance products to run under a different HA cluster than that for NNMi. In this case, you must manage the NNM iSPI Performance products' dependency on NNMi manually.

NNMi with an Oracle database scenario

If your NNMi implementation uses Oracle for the main NNMi database, the Oracle database should be on a separate server, as shown in [Figure 15](#), for performance reasons. Therefore, you must configure two HA resource groups within the NNMi HA cluster:

- The NNMi HA resource group includes the NNMi nodes and a shared disk for NNMi data that is not stored in the Oracle database.
- The Oracle HA resource group contains the Oracle database server and the database disk.

Figure 15 HA for NNMi with an Oracle Database

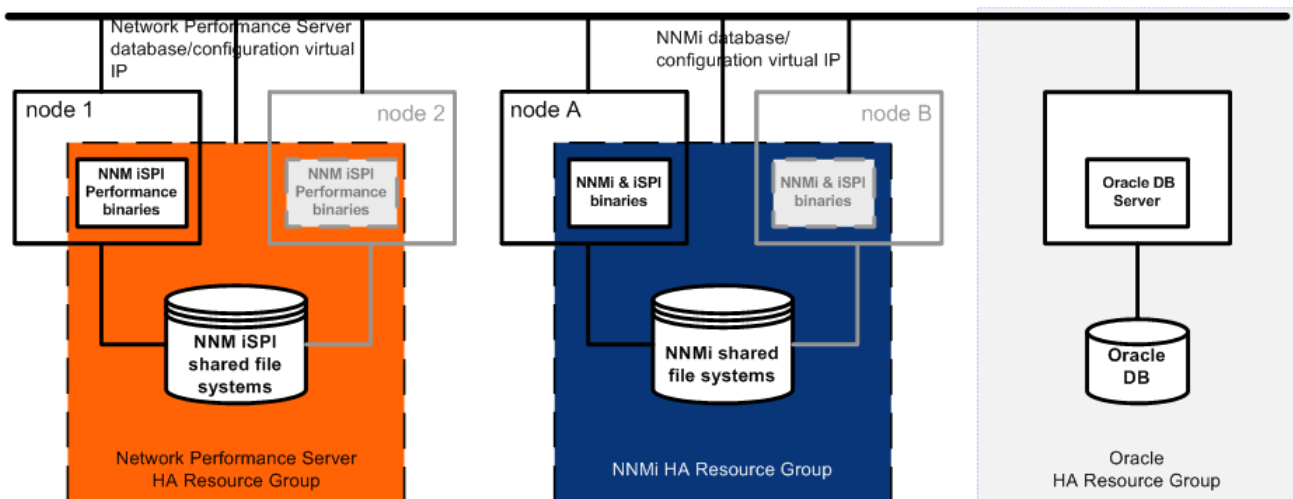


For information about how to implement this scenario, see [Configuring NNMi for HA in an Oracle Environment](#) on page 216 and [Configuring NNM iSPIs for HA](#) on page 215.

NNMi with an Oracle database and NNM iSPI Performance products on a dedicated server scenario

If your NNMi implementation uses Oracle for the main NNMi database and you are running any of the NNM iSPI Performance products on a dedicated server, you can configure three HA resource groups within the NNMi HA cluster, as shown in [Figure 16](#).

Figure 16 HA for NNMi with an Oracle Database and NNM iSPI Performance on a Dedicated Server



For information about how to implement this scenario, see [Configuring NNMi for HA in an Oracle Environment](#) on page 216 and [Configuring NNMi iSPIs for HA](#) on page 215.

Manpages

With regard to HA configuration, the NNMi manpages contain the following topics:

- nnm-ha
- nnmhaconfigure.ovpl
- nnmhaunconfigure.ovpl
- nnmhadisk.ovpl
- nnmhaclusterinfo.ovpl
- nnmhastartrg.ovpl
- nnmhastoprg.ovpl

On the Windows operating system, these manpages are available as text files.

Configuring HA

This section describes the procedures for configuring a new HA configuration for NNMi. It contains the following topics:

- [Configuring NNMi Certificates for HA](#) on page 209
- [Configuring NNMi for HA](#) on page 209
- [Configuring NNM iSPIs for HA](#) on page 215
- [Configuring NNMi for HA in an Oracle Environment](#) on page 216

Configuring NNMi Certificates for HA

The NNMi installation process configures a self-signed certificate for secure communications between the NNMi console and the NNMi database. The process for configuring NNMi for HA correctly shares the self-signed certificate among the primary and secondary cluster nodes. You do not need to take any extra steps to use the default certificate with NNMi running under HA.

If you want to use a different self-signed certificate or a Certificate Authority (CA)-signed certificate for NNMi communications, you need to do some additional work. After obtaining the new certificate, complete the steps shown in [Configuring High Availability for a New Certificate](#) on page 105. You can complete this procedure before or after configuring NNMi for HA.

Configuring NNMi for HA

The two distinct phases of configuring NNMi for HA are as follows:

- 1 Copy the NNMi data files to the shared disk.
 - Do this task on the primary node, as described in [step 1](#) through [step 12](#) of [Configuring NNMi on the Primary Cluster Node](#) on page 211.
- 2 Configure NNMi to run under HA.
 - Do this task on the primary node, as described in [step 13](#) through [step 17](#) of [Configuring NNMi on the Primary Cluster Node](#) on page 211.
 - Do this task on the secondary node, as described in [Configuring NNMi on the Secondary Cluster Nodes](#) on page 214.

Designate one HA cluster node as the primary NNMi management server. This is the node you expect to be active most of the time. Configure the primary node, and then configure all other nodes in the HA cluster as secondary nodes.



You *cannot* configure NNMi for HA simultaneously on multiple cluster nodes. After the HA configuration process is completed on one cluster node, proceed with the HA configuration on the next node, and so forth until NNMi is configured for HA on all nodes in the cluster environment.



During failover, the NNMi console is unresponsive. After failover completes, NNMi users must sign in to continue their NNMi console sessions.

NNMi HA Configuration Information

The HA configuration script collects information about the NNMi HA resource group. [Table 16](#) lists the information that you will need for configuring the primary node. Gather this information before you begin the configuration procedure.

Table 16 NNMi HA Primary Node Configuration Information

HA Configuration Item	Description
HA resource group	The name of the resource group for the HA cluster that contains NNMi. For example: nnmtest1
Virtual host short name	The short name for the virtual host. This hostname must map to the virtual IP address for the HA resource group. The <code>nslookup</code> command must be able to resolve the virtual host short name and the virtual IP address. NOTE: If NNMi is unable to resolve the virtual host short name or the virtual host IP address, the HA configuration script could leave the system in an unstable state. Therefore, HP recommends that you implement a secondary naming strategy (such as entering the information in the <code>%SystemRoot%\system32\drivers\etc\hosts</code> file on the Windows operating system or <code>/etc/hosts</code> file on UNIX operating systems) in case DNS is not available during NNMi HA configuration.
Virtual host netmask	The subnet mask that is used with the virtual host IP address, which must be an IPv4 address.
Virtual host network interface	The network interface on which the virtual host IP address is running. For example: <ul style="list-style-type: none">• <i>Windows:</i> Local Area Connection• <i>HP-UX:</i> lan0• <i>Linux:</i> eth0• <i>Solaris:</i> bge0
Shared file system type	The type of shared disk configuration being used for the HA resource group. Possible values are: <ul style="list-style-type: none">• <code>disk</code>—The shared disk is a physically attached disk that uses a standard file system type. The HA configuration script can configure the shared disk. For more information, see the File system type entry in this table.• <code>none</code>—The shared disk uses a configuration other than that described for the <code>disk</code> option, such as SAN or NFS. After running the HA configuration script, configure the shared disk as described in Configuring the Shared Disk on page 219.

Table 16 NNMi HA Primary Node Configuration Information

HA Configuration Item	Description
File system type	<p>(UNIX only) The file system type of the shared disk (if the shared file system type is <code>disk</code>). The HA configuration scripts pass this value to the HA product so that it can determine how to validate the disk.</p> <p>HP has tested the following shared disk formats:</p> <ul style="list-style-type: none"> • <i>Windows</i>: Basic (see A Note about Shared Disk Configuration on Windows Server on page 219) • <i>HP-UX</i>: vxfs • <i>Linux</i>: lvm2 for Serviceguard; ext2 for VCS • <i>Solaris</i>: vxfs <p>HP expects that other formats for a physically attached disk work for this HA implementation, and HP provides limited support in analyzing NNMi HA systems with other physically attached disks that use a standard file system type.</p> <p>NOTE: If you use a shared disk format that HP has not tested, review the resource group configuration to verify that the values inserted by the NNMi HA configuration scripts are correct.</p>
Disk group	<p>(UNIX only) The name of the disk group for the NNMi shared file system. This name is based on the name of the HA resource group.</p> <p>For example: <code>nnmtest1-dg</code></p>
Volume group	<p>(UNIX only) The name of the volume group for the NNMi shared file system. This name is based on the name of the HA resource group.</p> <p>For example: <code>nnmtest1-vol</code></p>
Mount point	<p>The directory location for mounting the NNMi shared disk. This mount point must be consistent between systems. (That is, each node must use the same name for the mount point.) For example:</p> <ul style="list-style-type: none"> • <i>Windows</i>: <code>S:\</code> • <i>UNIX</i>: <code>/nnmmount</code>

Configuring NNMi on the Primary Cluster Node

Complete the following procedure on the primary cluster node.



If you are using Oracle for the main NNMi database, see [Configuring NNMi for HA in an Oracle Environment](#) on page 216 first.

- 1 Verify that the system meets all of the requirements specified in [Prerequisites to Configuring NNMi for HA](#) on page 202.
- 2 If you have not already done so, install NNMi (including the latest consolidated patch, if any), and then verify that NNMi is working correctly.
- 3 If you expect to run any NNM iSPIs on this NNMi management server, see [Configuring NNM iSPIs for HA](#) on page 215 before continuing with this procedure.

- 4 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 [NNMi Backup and Restore Tools](#) on page 255.

- 5 Back up the NNMi license file by copying the following file to another location:

- *Windows:* %AUTOPASS_HOME%\data\LicFile.txt

To determine the value of %AUTOPASS_HOME%, examine the system environment variables for the computer.

- *UNIX:* /var/opt/OV/HPOvLIC/LicFile.txt

- 6 Define the disk device group (and logical volume), consisting of at least one shared disk for the NNMi HA resource group. For example:

- *MSFC or MSCS:* Use Disk Management to configure the disk mount point and format the disk.

- *Serviceguard:*

Use LVM commands such as `pvcreate`, `vgcreate`, and `lvcreate` to initialize the disk, create the volume group, and create the logical volume.

- *VCS:*

Use the Symantec Veritas Storage Foundation as described here:

Use `vxdiskadm` to add and initialize the disk.

Use `vxassist make` to allocate disks by space.

```
mkfs -F vxfs /dev/vx/dsk/<disk_group>/<logical_volume_group>
```

For UNIX operating systems, a reference web site is:

<http://www.unixguide.net/unixguide.shtml>

- 7 Create the directory mount point for the shared disk (for example, `S:\` or `/nnmmount`):

- *Windows:* Use Windows Explorer and Disk Management.

- *UNIX:* Verify that the shared disk directory mount point has been created with `root` as the user, `sys` as the group, and the permissions set to 555. For example:

```
ls -l /nnmmount
```

- 8 Mount the shared disk. For example:

- *Windows:* Use Disk Management.

- *HP-UX or Linux for Serviceguard:*

```
mount /dev/<disk_group>/<logical_volume> /nnmmount
```

- *Linux or Solaris for VCS:*

```
vxfs: mount /dev/vx/dsk/<disk_group>/<volume_group> /nnmmount
```

- 9 Stop NNMi:

```
ovstop -c
```



If NNMi is already installed on a node that you will include in this HA resource group, also run `ovstop -c` on that node at this time.

10 Copy the NNMi database to the shared disk:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM \  
-to <HA_mount_point>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM \  
-to <HA_mount_point>
```



To prevent database corruption, run this command (with the `-to` option) only one time. For information about alternatives, see [Re-Enable NNMi for HA after All Cluster Nodes are Unconfigured](#) on page 236.

11 Unmount the shared disk:

- *Windows:* Use Windows Explorer and Disk Management.

- *UNIX:* `umount <HA_mount_point>`

12 (UNIX only) Deactivate the disk group:

```
vgchange -a n <disk_group>
```

13 Verify that NNMi is not running:

```
ovstop -c
```

14 Configure the NNMi HA resource group:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl NNM
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl NNM
```

[Table 16](#) on page 210 describes the information that this command requests.

15 (UNIX only) By default NNMi starts in the locale of the user who ran the `nmhaconfigure.ovpl` command. To change the NNMi locale, run the following command:

```
$NnmInstallDir/misc/nnm/ha/ nmhaclusterinfo.ovpl \  
-config NNM -set HA_LOCALE <locale>
```

16 In [step 14](#), what value did you specify for the shared file system type (as described for [Shared file system type](#) and [File system type](#) in [Table 16](#) on page 210)?

- For type `disk`, the `nmhaconfigure.ovpl` command configured the shared disk. Continue with [step 17](#).
- For type `none`, configure the shared disk as described in [Configuring the Shared Disk](#) on page 219; then continue with [step 17](#).

17 Start the NNMi HA resource group:

- *Windows:*
`%NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM \
<resource_group>`
- *UNIX:*
`$NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM \
<resource_group>`

If NNMi does not start correctly, see [Troubleshooting the HA Configuration](#) on page 234.



Now that NNMi is running under HA, *do not* use the `ovstart` and `ovstop` commands for normal operation. Use these commands only when instructed to do so for HA maintenance purposes.

Configuring NNMi on the Secondary Cluster Nodes

Complete the following procedure on one secondary cluster node at a time.

- 1 If you have not already done so, complete the procedure for [Configuring NNMi on the Primary Cluster Node](#) on page 211.
- 2 Verify that the system meets all of the requirements specified in [Prerequisites to Configuring NNMi for HA](#) on page 202.
- 3 If you have not already done so, install NNMi (including the latest consolidated patch, if any), and then verify that NNMi is working correctly.
- 4 Install the NNM iSPIs that you installed in [step 3 of Configuring NNMi on the Primary Cluster Node](#) on page 211.
- 5 Stop NNMi:

```
ovstop -c
```

- 6 Create a mount point for the shared disk (for example, `S:\` or `/nnmmount`).



This mount point must use the same name as the mount point you created in [step 7 of the procedure Configuring NNMi on the Primary Cluster Node](#).

- 7 Configure the NNMi HA resource group:
 - *Windows:* `%NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl NNM`
 - *UNIX:* `$NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl NNM`

Supply the HA resource group name when the command requests this information.

8 Verify that the configuration was successful:

- *Windows:*

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

- *UNIX:*

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

The command output lists all configured nodes for the specified HA resource group.

9 Optionally, test the configuration by taking the resource group on the primary node offline and then bringing the resource group on the secondary node online.

Configuring NNM iSPIs for HA

If you expect to run any NNM iSPIs on the NNMi management server, read this section before configuring NNMi to run under HA.

NNM iSPI for Metrics, NNM iSPI for QA, and NNM iSPI for Traffic

The NNM iSPI Performance products (NNM iSPI for Metrics, NNM iSPI for QA, and NNM iSPI for Traffic) can be installed on the NNMi management server or on a dedicated server, but not on a combination of these two options.

- If the NNM iSPI Performance products will be located on the NNMi management server, install the products before configuring NNMi to run under HA.
- If the NNM iSPI Performance products will be located on a dedicated server, configure NNMi to run under HA before installing the products. During the NNM iSPI installation process, supply the NNMi resource group virtual hostname as the NNMi management server name.

NNM iSPI for MPLS, NNM iSPI for IP Multicast, and NNM iSPI for IP Telephony

The NNM iSPI for MPLS, NNM iSPI for IP Multicast, and NNM iSPI for IP Telephony can be installed on the NNMi management server only. Install these products before configuring NNMi to run under HA.

For information about configuring the NNM iSPIs to run under HA, see the appropriate NNM iSPI deployment guide.

NNM iSPI Network Engineering Toolset Software and NNMi Running under HA

The NNM iSPI Network Engineering Toolset Software SNMP trap analytics and Microsoft Visio export functionality are automatically installed with NNMi. No extra work is needed to run these tools under HA.

The NNM iSPI NET diagnostics server cannot be included in the NNMi HA resource group. Do not install this component on the NNMi management server. To run the NNM iSPI NET diagnostics server on a system that is outside the NNMi HA resource group, follow these steps:

- 1 Completely configure the NNMi HA resource group.

- 2 Install the NNM iSPI NET diagnostics server on a system that is outside the NNMi HA resource group. During the NNM iSPI NET diagnostics server installation process, supply the NNMi resource group virtual hostname as the NNM Server Hostname.

For more information, see the *NNM iSPI Network Engineering Toolset Software Planning and Installation Guide*.

If the NNM iSPI NET diagnostics server is already installed on an NNMi management server that will run under HA, uninstall the NNM iSPI NET diagnostics server before configuring NNMi to run under HA.



Uninstalling the NNM iSPI NET diagnostics server removes all existing reports.



It might be possible to save existing reports, as described here, but the following procedure is untested:

- 1 Use MySQL Workbench to perform a backup of the existing `nnminet` database. MySQL Workbench is available in the downloads area at **dev.mysql.com**.
- 2 Uninstall the NNM iSPI NET diagnostics server.
- 3 Configure NNMi to run under HA.
- 4 Install the NNM iSPI NET diagnostics server on a separate system.
- 5 Before running any flows, use MySQL Workbench to recover the `nnminet` database onto the new installation.

Configuring NNMi for HA in an Oracle Environment

This sections presents a high-level overview of the process for configuring NNMi with an Oracle database to run under HA. The number of possible Oracle configurations is large, and the configuration process can vary according to the Oracle release. For the most accurate information about configuring Oracle to run under HA and creating an NNMi dependency on the Oracle HA resource group, see the HA product documentation. You can also go to the Oracle web site (**www.oracle.com**) for information about the appropriate Oracle configuration for your HA product.

NNMi with Oracle HA Configuration Information

When Oracle and NNMi both run under HA, each product must be in a separate HA resource group. The Oracle resource group must be fully started before the NNMi resource group starts. If both resource groups are in the same HA cluster, you can modify the cluster configuration to set resource group ordering. If the resource groups are in different HA clusters, make sure the NNMi resource group dependency on the Oracle resource group is met.

The NNMi resource group must include a shared disk for the NNMi data that is not stored in the Oracle database.

Configuring NNMi with Oracle for HA

- 1 If you plan to run Oracle under HA, complete that configuration first.
- 2 Create an empty Oracle database instance for NNMi.

- 3 On the primary NNMi node, install NNMi (including the latest consolidated patch, if any). During installation, do the following:
 - a Select the **Oracle** database type, and then select **Primary Server Installation**.
 - b Specify the virtual IP address or hostname for the Oracle HA resource group (if applicable).
- 4 On the primary NNMi node, configure NNMi to run under HA as described in [Configuring NNMi on the Primary Cluster Node](#) on page 211.
- 5 Set up the NNMi dependency on the Oracle HA resource group.
For specific instructions, see the HA product documentation.
- 6 On the secondary NNMi node, install NNMi (including the latest consolidated patch, if any). During installation, do the following:
 - Select the **Oracle** database type, and then select **Secondary Server Installation**.
 - Specify the virtual IP address or hostname for the Oracle HA resource group (if applicable).
- 7 On the secondary NNMi node, configure NNMi to run under HA described in [Configuring NNMi on the Secondary Cluster Nodes](#) on page 214.
- 8 For each additional secondary NNMi node, repeat [step 6](#) and [step 7](#).

Shared NNMi Data

This implementation of NNMi running under HA requires the use of a separate disk for sharing files between all NNMi nodes in the HA cluster.



NNMi implementations that use Oracle as the primary database also require the use of a separate disk for shared data.

Data on the NNMi Shared Disk

This section lists the NNMi data files that are maintained on the shared disk when NNMi is running under HA.

The locations are mapped to the shared disk location as follows:

- *Windows:*
 - %NnmInstallDir% maps to %HA_MOUNT_POINT%\NNM\installDir
 - %NnmDataDir% maps to %HA_MOUNT_POINT%\NNM\dataDir
- *UNIX:*
 - \$NnmInstallDir maps to \$HA_MOUNT_POINT/NNM/installDir
 - \$NnmDataDir maps to \$HA_MOUNT_POINT/NNM/dataDir

The directories that are moved to the shared disk are as follows:

- *Windows:*
 - %NnmDataDir%\shared\nnm\databases\Postgres
The embedded database; not present when using an Oracle database.
 - %NnmDataDir%\log\nnm
The NNMi logging directory.
 - %NnmDataDir%\shared\nnm\databases\eventdb
The pmd events database.
 - %NnmInstallDir%\nonOV\jboss\nms\server\nms\data
The transactional store used by ovjboss.
- *UNIX:*
 - \$NnmDataDir/shared/nnm/databases/Postgres
The embedded database; not present when using an Oracle database.
 - \$NnmDataDir/log/nnm
The NNMi logging directory.
 - \$NnmDataDir/shared/nnm/databases/eventdb
The pmd events database.
 - \$NnmInstallDir/nonOV/jboss/nms/server/nms/data
The transactional store used by ovjboss.

The `nnmhadisk.ovpl` command copies these files to and from the shared disk. Run this command as the instructions in this chapter indicate. For a summary of the command syntax, see the `nnm-ha` manpage.

Replication of Configuration Files

The NNMi HA implementation uses file replication to maintain copies of the NNMi configuration files on all NNMi nodes in the HA cluster. By default, the NNMi command `nmdatareplicator.ovpl` manages file replication. This command copies the NNMi configuration files from the active node to a passive node during the failover process. The `nmdatareplicator.conf` file specifies the NNMi folders and files that are included in data replication.

For information about the data replication process, see the `nnm-ha` manpage.

Configuring the Shared Disk

If the shared disk is of a format that HP has tested (as listed in [Table 16](#) on page 210), the HA configuration script prepares the shared disk, and you can safely ignore this section.

If the shared disk uses a non-tested configuration, such as SAN or NFS, you must prepare the disk manually. Enter the value `none` for the file system type during HA configuration, and then modify the disk configuration file according to the product documentation. For example:

- **MSFC or MSCS:** Use Cluster Administrator (`cluadmin.exe`) or the `cluster.exe` command to add resources to the resource group.
- **ServiceGuard:**
 - **HP-UX:**
`/etc/cmcluster/<resource_group>/<resource_group>.cntl`
 - **Linux:**
`/usr/local/cmcluster/conf/<resource_group>/<resource_group>.cntl`
- **VCS:** Add disk entries and links to the HA configuration file by using the `/opt/VRTSvcs/bin/hares` command.

A Note about Shared Disk Configuration on Windows Server

According to Microsoft Knowledge Base article 237853, dynamic disks are not supported for clustering with Windows Server 2008 or Windows Server 2003. To ensure the correct disk configuration, review the information on the following web sites:

- <http://support.microsoft.com/kb/237853>
- http://www.petri.co.il/difference_between_basic_and_dynamic_disks_in_windows_xp_2000_2003.htm

Licensing NNMi in an HA Cluster

For NNMi running under HA, the NNMi license is tied to the virtual IP address of the cluster. Therefore, an HA cluster requires only one NNMi license.

To correctly install the NNMi license for an HA cluster, perform these steps on the active NNMi cluster node:

- 1 At a command prompt, enter the following command:
 - *Windows:* `%NnmInstallDir%\bin\nnmlicense.ovpl NNM -g`
 - *UNIX:* `$NnmInstallDir/bin/nnmlicense.ovpl NNM -g`
- 2 In the **License Password** dialog box, click **Request License**.
- 3 Follow the instructions on the screen to obtain the permanent license password for the virtual IP address of the HA cluster.
- 4 At a command prompt, enter the following command to update your system and to store your license data files:
 - *Windows:*
`%NnmInstallDir%\bin\nnmlicense.ovpl NNM -f <license_file>`
 - *UNIX:*
`$NnmInstallDir/bin/nnmlicense.ovpl NNM -f <license_file>`
- 5 Update the file `licenses.txt` in the NNM directory on the shared disk (for example `S:\NNM\licenses.txt` or `/nnmount/NNM/licenses.txt`) with the new information from the license file on the active node, which is in the following location:
 - *Windows:* `%AUTOPASS_HOME%\data\LicFile.txt`
To determine the value of `%AUTOPASS_HOME%`, examine the system environment variables for the computer.
 - *UNIX:* `/var/opt/OV/HPOvLIC/LicFile.txt`Do one of the following:
 - If this file exists on the shared disk, append the new license keys in `LicFile.txt` on the active node to `licenses.txt` on the shared disk.
 - If this file does not exist on the shared disk, copy `LicFile.txt` from the active node to `licenses.txt` in the NNM directory on the shared disk.

Maintaining the HA Configuration

Maintenance Mode

When you need to apply NNMi patches, put the NNMi HA resource group into maintenance mode to prevent failover during the process. When the NNMi HA resource group is in maintenance mode, you (or an installation script) can run the `ovstop` and `ovstart` commands as needed on the primary (active) cluster node.



Never run the `ovstart` or `ovstop` commands on a secondary (backup) cluster node.

Putting an HA Resource Group into Maintenance Mode

Putting an HA resource group into maintenance mode disables HA resource group monitoring. When an HA resource group is in maintenance mode, stopping and starting the products in that HA resource group do not cause failover.

To put an HA resource group into maintenance mode, on the active cluster node, create the following file:

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

The file can be empty.

Removing an HA Resource Group from Maintenance Mode

Taking an HA resource group out of maintenance mode re-enables HA resource group monitoring. Stopping the products in that HA resource group causes the HA resource group to fail over to a passive cluster node.

To remove an HA resource group from maintenance mode, delete the maintenance file from the node that was the active cluster node before maintenance was initiated. This file is described in [Putting an HA Resource Group into Maintenance Mode](#).

Maintaining NNMi in an HA Cluster

Starting and Stopping NNMi

While NNMi is running under HA, *do not* use the `ovstart` and `ovstop` commands unless instructed to do so for HA maintenance purposes. For normal operation, use the NNMi-provided HA commands or the appropriate HA product commands for starting and stopping resource groups.

Changing NNMi Hostnames and IP Addresses in a Cluster Environment

A node in a cluster environment can have more than one IP address and hostname. If a node becomes a member of another subnet, you might need to change its IP addresses. As a result, the IP address or fully-qualified domain name might change.

For example, on UNIX systems, the IP address and the related hostname are generally configured in one of the following:

- `/etc/hosts`
- Domain Name Service (DNS)
- Network Information Service (NIS on HP-UX or Linux, NIS+ on Solaris)

NNMi also configures the hostname and IP address of the management server for the managed node in the NNMi database.

If you are moving from a non-name-server environment to a name-server environment (that is, DNS or BIND), make sure that the name server can resolve the new IP address.

Hostnames work within IP networks to identify a managed node. While a node may have many IP addresses, the hostname is used to pinpoint a specific node. The system hostname is the string returned when you use the `hostname` command.

To change the virtual hostname or IP address of the management server, perform these steps on the active NNMi cluster node:

- 1 At a command prompt, enter the following command:
 - *Windows:* `%NnmInstallDir%\bin\nnmlicense.ovpl NNM -g`
 - *UNIX:* `$NnmInstallDir/bin/nnmlicense.ovpl NNM -g`
 - 2 In the **License Password** dialog box, click **Request License**.
 - 3 Follow the instructions on the screen to obtain the permanent license password for the new virtual IP address of the HA cluster.
 - 4 At a command prompt, enter the following command to update your system and to store your license data files:
 - *Windows:*
`%NnmInstallDir%\bin\nnmlicense.ovpl NNM -f <license_file>`
 - *UNIX:*
`$NnmInstallDir/bin/nnmlicense.ovpl NNM -f <license_file>`
 - 5 Update the file `licenses.txt` in the NNM directory on the shared disk (for example `S:\NNM\licenses.txt` or `/nnmount/NNM/licenses.txt`) with the new information from the license file on the active node, which is in the following location:
 - *Windows:* `%AUTOPASS_HOME%\data\LicFile.txt`
To determine the value of `%AUTOPASS_HOME%`, examine the system environment variables for the computer.
 - *UNIX:* `/var/opt/OV/HPOvLIC/LicFile.txt`
- Do one of the following:
- If this file exists on the shared disk, append the new license keys in `LicFile.txt` on the active node to `licenses.txt` on the shared disk.
 - If this file does not exist on the shared disk, copy `LicFile.txt` from the active node to `licenses.txt` in the NNM directory on the shared disk.
- 6 Put the HA resource group into maintenance mode as described in [Putting an HA Resource Group into Maintenance Mode](#) on page 221.

7 Stop NNMi:

```
ovstop -c
```

8 Change the IP address or node name of the NNMi management server:

- a In the `ov.conf` file, edit the `NNM_INTERFACE` entry to be the new hostname or IP address.
- b In the `ovspmd.auth` file, edit any lines containing the old hostname to contain the new hostname.

The `ov.conf` and `ovspmd.auth` files are available in the following location:

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

9 If you changed the node name of the NNMi management server, set NNMi to use the new fully-qualified domain name of the NNMi management server with the `nmmsetofficialfqdn.ovpl` command. For example:

```
nmmsetofficialfqdn.ovpl newnnmi.servers.example.com
```

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

10 Set the cluster configuration:

- a 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>
```

- b Change the cluster configuration to use the new IP address:

— *MSFC or MSCS:*

In Cluster Administrator, open `<resource_group>`.

Double-click `<resource_group>-ip`, select **Parameters**, and then enter the new IP address.

— *Serviceguard:*

On the active HA cluster node, edit the `<resource_group>.cntl` file to replace `IP[0]=<old_IP_address>` with `IP[0]=<new_IP_address>`. Then use `cmapplyconf` to update all other systems.

```
HP-UX: /etc/cmcluster/<resource_group>/<resource_group>.cntl
```

```
Linux: /usr/local/cmcluster/conf/<resource_group>/  
<resource_group>.cntl
```

— *VCS:*

```
$NnmInstallDir/misc/nnm/ha/nnmhargconfigure.ovpl NNM \  
<resource_group> -set_value <resource_group>-ip \  
Address <new_IP_address>
```

- c Start the NNMi HA resource group:

- *Windows:*

- ```
%NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM \
<resource_group>
```

- *UNIX:*

- ```
$NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM \  
<resource_group>
```

- 11 Verify that NNMi started correctly:

- ```
ovstatus -c
```

- All NNMi services should show the state RUNNING.

- 12 Take the HA resource group out of maintenance mode as described in [Removing an HA Resource Group from Maintenance Mode](#) on page 221.

## Stopping NNMi Without Causing Failover

When you need to perform NNMi maintenance, you can stop NNMi on the active cluster node without causing failover to a currently passive node. Follow these steps on the active cluster node:

- 1 Put the HA resource group into maintenance mode as described in [Putting an HA Resource Group into Maintenance Mode](#) on page 221.
- 2 Stop NNMi:

- ```
ovstop -c
```

Restarting NNMi after Maintenance

If you have stopped NNMi in the manner that prevents failover, follow these steps to restart NNMi and HA monitoring:

- 1 Start NNMi:

- ```
ovstart -c
```

- 2 Verify that NNMi started correctly:

- ```
ovstatus -c
```

- All NNMi services should show the state RUNNING.

- 3 Take the HA resource group out of maintenance mode as described in [Removing an HA Resource Group from Maintenance Mode](#) on page 221.

Maintaining Add-on NNM iSPIs in an NNMi HA Cluster

The NNM iSPIs are closely linked to NNMi. When add-on NNM iSPIs are installed on the nodes in the NNMi HA cluster, use the NNMi HA cluster maintenance procedures as written.

Unconfiguring HA

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 deployment guide 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, remove the resource group-specific files:

- *MSFC or MSCS:*

In Windows Explorer, delete all files in 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 deployment guide for each NNM iSPI.

- 7 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.

8 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.

9 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.

10 On the active node, remove the resource group-specific files:

- *MSFC or MSCS:*

In Windows Explorer, delete all files in 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>/*
```

11 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 the next procedure), and then use the HA product commands to unconfigure the disk group and volume group.

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/vgnnm/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 /nnmmount`

- 7 (UNIX only) Deactivate the disk group:

```
vgchange -a n <disk_group>
```

- 8 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, follow these steps:
 - a Put the HA resource group into maintenance mode as described in [Putting an HA Resource Group into Maintenance Mode](#) on page 221.
 - b Stop NNMi:

```
ovstop -c
```
 - c Back up the shared disk by performing a disk copy.
 - d *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](#) on page 255.
 - e Apply the appropriate NNMi patches to the system.
 - f Start NNMi:

```
ovstart -c
```
 - g Verify that NNMi started correctly:

```
ovstatus -c
```

All NNMi services should show the state `RUNNING`.
- 3 On each passive node, follow these steps:
 - a Put the HA resource group into maintenance mode as described in [Putting an HA Resource Group into Maintenance Mode](#) on page 221.
 - b Apply the appropriate patches to the system.

 Never run the `ovstart` or `ovstop` commands on a secondary (backup) cluster node.

 - c Take the HA resource group out of maintenance mode as described in [Removing an HA Resource Group from Maintenance Mode](#) on page 221.
- 4 On the active node, take the HA resource group out of maintenance mode as described in [Removing an HA Resource Group from Maintenance Mode](#) on page 221.

Upgrading NNMi under HA from NNMi 8.1x to NNMi 9.00

To upgrade from NNMi 8.1x under HA to NNMi 9.00 under HA, unconfigure NNMi 8.1x from HA, install NNMi 9.00 to get the updated database schema, and then reconfigure NNMi for HA. Follow these steps:

- 1 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.
- 2 Determine which node in the NNMi 8.1x 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
```

- 3 On each passive node, unconfigure NNMi 8.1x and install NNMi 9.00:

- a Unconfigure any add-on NNM iSPIs from the HA cluster.

For information, see the deployment guide for each NNM iSPI.

- b 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 a](#) until this command output includes only the active node hostname.

- c 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.

- d Remove the resource group-specific files:
 - *MSFC* or *MSCS*:
In Windows Explorer, delete all files in 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>/*
```
- e Install NNMi 9.00 and the latest NNMi consolidated patch (if any).



At this point the passive nodes are not functional NNMi management servers. Do not reconfigure the passive nodes into the HA cluster until the active node has also been upgraded to NNMi 9.0x Patch 1 and reconfigured into the HA cluster.

- 4 On the active node, unconfigure NNMi 8.1x:
 - a Disable HA resource group monitoring by creating the following maintenance file:
 - *Windows*: %NnmDataDir%\hacluster*<resource_group>*\maintenance
 - *UNIX*: \$NnmDataDir/hacluster/*<resource_group>*/maintenance

The file can be empty.
 - b Stop NNMi:

```
ovstop -c
```



To prevent data corruption, make sure that no instance of NNMi is running and accessing the shared disk.

- c Back up the NNMi database that is on the shared disk:

```
nnmbackup.ovpl -type offline -target <backup_directory>
```

For more information about this command, see [NNMi Backup and Restore Tools](#) in the *NNMi Deployment Guide* for NNMi 8.1x. (The `nnmbackup.ovpl` command changed at NNMi version 8.13. Be sure to reference the information for your version of NNMi.)
- d Copy the NNMi files from the shared disk to the node:
 - *Windows*:

```
%NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl \
```

```
-from <HA_mount_point>
```
 - *UNIX*:

```
$NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl \
```

```
-from <HA_mount_point>
```

- e Move aside all NNMi files and directories from the shared disk:
 - *Windows:* Use Windows Explorer to move all files under the shared disk mount point (%HA_MOUNT_POINT%, which is, for example S:\) to another location such as %HA_MOUNT_POINT%\8_10.

- *UNIX:*

```
cd $HA_MOUNT_POINT
mkdir 8_10
mv NNM 8_10/.
```

Also move the directories for any NNM iSPIs to the 8_10 directory.

- f 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>
```

- g Unconfigure NNMi from the HA cluster:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl \
<resource_group>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl \
<resource_group>
```

This command removes access to the shared disk but does not unconfigure the disk group or the volume group.

- h Remove the resource group-specific files:

- *MSFC or MSCS:*

In Windows Explorer, delete all files in 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>/*
```

- i Delete the maintenance file:

- *Windows:* %NnmDataDir%\hacluster*<resource_group>*\maintenance

- *UNIX:* \$NnmDataDir/hacluster/*<resource_group>*/maintenance

- 5 On the node that was active before unconfiguring NNMi 8.1x from HA, configure NNMi 9.00:
 - a Install NNMi 9.00.

The update process starts NNMi to update the NNMi database schema.
 - b Install the latest NNMi consolidated patch.
 - c Verify that NNMi 9.00 is running correctly.
 - d Delete the NNMi 8.1x data from the shared disk:
 - *Windows*: Use Windows Explorer to delete the %HA_MOUNT_POINT%\8_10 folder.
 - *UNIX*:

```
cd $HA_MOUNT_POINT
rm -rf ./8_10
```
 - e Follow the steps in [Configuring NNMi on the Primary Cluster Node](#) on page 211.

You do not need to do the following:

 - Define a disk device group and logical volume.
 - Create a mount point for the shared disk.
 - Configure the shared disk.
- 6 On each node that was passive before unconfiguring NNMi 8.1x from HA, configure NNMi 9.00:
 - f On the first secondary node, follow the steps in [Configuring NNMi on the Secondary Cluster Nodes](#) on page 214.

You do not need to create a mount point for the shared disk.

Troubleshooting the HA Configuration

General HA Troubleshooting

The topics in this section apply to HA configuration for all HP NNM i-suite Software products.

Error: Wrong Number of Arguments

The name of the product Perl module is a required parameter to most of the NNMi HA configuration commands.

- For NNMi use the value `NNM`.
- To determine what value to use for an NNM iSPI, see the deployment guide for that NNM iSPI.

Product Startup Times Out (Solaris)

One or more of the `/var/adm/messages*` files contains a message similar to the following example:

```
VCS ERROR V-16-1-13012 Thread(...) Resource(<resource_group>-app):  
online procedure did not complete within the expected time.
```

This message indicates that the product did not start completely within the Veritas timeout value. The NNMi-provided HA configuration scripts define this timeout to be 15 minutes.

To change the Veritas timeout value on the Solaris operating system, run the following commands in order:

```
/opt/VRTSvcs/bin/haconf -makerw  
/opt/VRTSvcs/bin/hares -modify <resource_group>-app OnlineTimeout <value in seconds>  
/opt/VRTSvcs/bin/haconf -dump -makero
```

Log Files on the Active Cluster Node Are Not Updating

This situation is normal. It occurs because the log files have been redirected to the shared disk.

For NNMi, review the log files in the location specified by `HA_NNM_LOG_DIR` in the `ov.conf` file.

Cannot Start the HA Resource Group on a Particular Cluster Node

If the `nmmhargconfigure.ovpl` command does not correctly start, stop, or switch the NNMi HA resource group, review the following information:

- *MSFC* or *MSCS*:
 - In Cluster Administrator, review the state of the resource group and underlying resources.
 - Review the Event Viewer log for any errors.
- *Serviceguard*:

Review the `<resource_group>.cntl.log` file and the syslog files for errors. The most common problems are leaving the system in a state where a resource cannot be added, for example, having a disk group misconfigured such that it cannot be activated.

 - *HP-UX*: `/etc/cmcluster/<resource_group>/<resource_group>.cntl.log`
 - *Linux*: `/usr/local/cmcluster/conf/<resource_group>/<resource_group>.cntl.log`
- *VCS*:
 - Run `/opt/VRTSvcs/bin/hares -state` to review the resource state.
 - For failed resources, review the `/var/VRTSvcs/log/<resource>.log` file for the resource that is failing. Resources are referenced by the agent type, for example: `IP*.log`, `Mount*.log`, and `Volume*.log`.

If you cannot locate the source of the problem, you can manually start the HA resource group by using the HA product commands:

- 1 Mount the shared disk.
- 2 Assign the virtual host to the network interface:
 - *MSFC* or *MSCS*:
 - Start Cluster Administrator.
 - Expand the resource group.
 - Right-click `<resource_group>ip`, and then click **Bring Online**.
 - *Serviceguard*:
 - *HP-UX*: Run `/usr/sbin/cmmodnet` to add the IP address.
 - *Linux*: Run `/usr/local/cmcluster/bin/cmmodnet` to add the IP address.
 - *VCS*: `/opt/VRTSvcs/bin/hares -online <resource_group>-ip \
-sys <local_hostname>`

3 Start the HA resource group. For example:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhastart.ovpl NNM \  
-start <resource_group>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhastart.ovpl NNM \  
-start <resource_group>
```

The return code 0 indicates that NNMi started successfully.

The return code 1 indicates that NNMi did not start correctly.

NNMi-Specific HA Troubleshooting

The topics in this section apply to HA configuration for NNMi only.

Re-Enable NNMi for HA after All Cluster Nodes are Unconfigured

When all NNMi HA cluster nodes have been unconfigured, the `ov.conf` file no longer contains any mount point references to the NNMi shared disk. To re-create the mount point reference without overwriting the data on the shared disk, follow these steps on the primary node:

1 If NNMi is running, stop it:

```
ovstop -c
```

2 Reset the reference to the shared disk:

- *Windows:*

```
%NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM \  
-setmount <HA_mount_point>
```

- *UNIX:*

```
$NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM \  
-setmount <HA_mount_point>
```

3 In the `ov.conf` file, verify the entries related to HA mount points.

For the location of the `ov.conf` file, see [NNMi HA Configuration Files](#) on page 239.

NNMi Does Not Start Correctly Under HA

Either of the following situations occurs:

- `ovspmd` does not start after NNMi is configured for HA. (`ovstatus` shows that `ovspmd` is NOT RUNNING.)
- `ovstart` or `ovstop` returns the following message:

```
ovstart: Must specify managers by name if running on OVW Client
```



While NNMi is running under HA, *do not* use the `ovstart` and `ovstop` commands unless instructed to do so for HA maintenance purposes.

The NNMi configuration points to a different system than where NNMi is running. To fix the problem, verify that the `ov.conf` file has appropriate entries for the following items:

- `NNM_INTERFACE=<virtual_hostname>`
- `NNM_HA_CONFIGURED=YES`
- `HA_RESOURCE_GROUP=<resource_group>`
- `HA_POSTGRES_DIR=<HA_mount_point>/NNM/dataDir/shared/nnm/databases/Postgres`
- `HA_EVENTDB_DIR=<HA_mount_point>/NNM/dataDir/shared/nnm/eventdb`
- `HA_NNM_LOG_DIR=<HA_mount_point>/NNM/dataDir/log`
- `HA_JBOSS_DATA_DIR=<HA_mount_point>/NNM/installDir/nonOV/jboss/nms/server/nms/data`
- `HA_MOUNT_POINT=<HA_mount_point>`

For the location of the `ov.conf` file, see [NNMi HA Configuration Files](#) on page 239.

nmsdbmgr Does Not Start after HA Configuration

This situation usually occurs as a result of starting NNMi immediately after running the `nnmhadisk.ovpl` command with the `-to` option. (The `nnmhaconfigure.ovpl` command has not been run.) In this case, the `HA_POSTGRES_DIR` entry in the `ov.conf` file specifies the location of the embedded database on the shared disk, but this location is not available to NNMi.

Run `nnmhaconfigure.ovpl`, and then complete the HA configuration.

pmd Does Not Start after HA Configuration

This situation usually occurs after a configuration error such as not setting up the shared disk correctly. The failure of the `pmd` process occurs when the `ovjboss` process does not fully start.

Review the following log file:

- *Windows:* `%HA_MOUNT_POINT%\NNM\dataDir\log\nnm\jbossServer.log`
- *UNIX:* `$HA_MOUNT_POINT/NNM/dataDir/log/nnm/jbossServer.log`

Disk Failover Does Not Occur

This situation can happen when the operating system does not support the shared disk. Review the appropriate HA product, operating system, and disk manufacturer documentation to determine whether these products can all work together.

If disk failure occurs, NNMi does not start on failover. Most likely, `nmsdbmgr` fails because the `HA_POSTGRES_DIR` directory does not exist. Verify that the shared disk is mounted and that the appropriate files are accessible.

Shared Disk Files Are Not Found on the Secondary Node after Failover

The most common cause of this situation is that the `nnmhadisk.ovpl` command was run with the `-to` option when the shared disk was not mounted. In this case, the data files are copied to the local disk, so the files are not available on the shared disk.

NNM iSPI-Specific HA Troubleshooting

For information about troubleshooting an NNM iSPI running under HA, see the deployment guide for that NNM iSPI.

HA Configuration Reference

NNMi HA Configuration Files

Table 17 lists the NNMi HA configuration files. These files apply to NNMi and add-on NNM iSPIs on the NNMi management server. These files are installed to the following location:

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

Table 17 NNMi HA Configuration Files

File Name	Description
ov.conf	Updated by the <code>nnmhaclusterinfo.ovpl</code> command to describe the NNMi HA implementation. NNMi processes read this file to determine the HA configuration.
nnmdatareplicator.conf	Used by the <code>nnmdatareplicator.ovpl</code> command to determine which NNMi folders and files are included in data replication from the active node to the passive nodes. If you implement a different method of replicating the NNMi configuration, see this file for a list of the data to include. For more information, see the comments in the file.

NNMi-Provided HA Configuration Scripts

Table 18 and Table 19 list the HA configuration scripts that are included with NNMi. The NNMi-provided scripts listed in Table 18 are convenience scripts that can be used to configure HA for any product that has a customer Perl module. If you prefer, you can use the HA product-provided commands to configure HA for NNMi.

On the NNMi management server, the NNMi-provided HA configuration scripts are installed to the following location:

- *Windows*: %NnmInstallDir%\misc\nnm\ha
- *UNIX*: \$NnmInstallDir/misc/nnm/ha

Table 18 NNMi HA Configuration Scripts

Script Name	Description
nnmhaconfigure.ovpl	Configures NNMi or an NNM iSPI for an HA cluster. Run this script on all nodes in the HA cluster.
nnmhaunconfigure.ovpl	Unconfigures NNMi or an NNM iSPI from an HA cluster. Optionally, run this script on one or more nodes in the HA cluster.
nnmhaclusterinfo.ovpl	Retrieves cluster information regarding NNMi. Run this script as needed on any node in the HA cluster.

Table 18 NNMi HA Configuration Scripts (cont'd)

Script Name	Description
nnmhadisk.ovpl	Copies NNMi and NNM iSPI data files to and from the shared disk. During HA configuration, run this script on the primary node. At other times, run this script per the instructions in this chapter.
nnmhastartrg.ovpl	Starts NNMi in an HA cluster. During HA configuration, run this script on the primary node.
nnmhastoprg.ovpl	Stops NNMi in an HA cluster. During HA unconfiguration, run this script on the primary node.

The NNMi-provided scripts listed in [Table 19](#) are used by the scripts listed in [Table 18](#) on page 239. Do not run the scripts listed in [Table 19](#) directly.

Table 19 NNMi HA Support Scripts

Script Name	Description
nnmdatareplicator.ovpl	Checks the <code>nnmdatareplicator.conf</code> configuration file for changes and copies files to remote systems.
nnmharg.ovpl	Starts, stops, and monitors NNMi in an HA cluster. For Serviceguard configurations, used by <code><resource_group>.cntl</code> . For VCS configurations, used by the VCS start, stop, and monitor scripts. (<code>nnmhargconfigure.ovpl</code> configures this usage.) Also used by <code>nnmhastartrg.ovpl</code> to enable and disable tracing.
nnmhargconfigure.ovpl	Configures HA resources and resource groups. Used by <code>nnmhaconfigure.ovpl</code> and <code>nnmhaunconfigure.ovpl</code> .
nnmhastart.ovpl	Starts NNMi in an HA cluster. Used by <code>nnmharg.ovpl</code> .
nnmhastop.ovpl	Stops NNMi in an HA cluster. Used by <code>nnmharg.ovpl</code> .
nnmhamonitor.ovpl	Monitors NNMi processes in an HA cluster. Used by <code>nnmharg.ovpl</code> .
nnmhamscs.vbs	Is a template for creating a script to start, stop, and monitor NNMi processes in a MSFC or MSCS HA cluster. The generated script is used by MSFC or MSCS and is stored in the following location: <code>%NmDataDir%\hacluster\<<resource_group>\hamscs.vbs</code>

NNMi HA Configuration Log Files

The following log files apply to the HA configuration for NNMi and add-on NNM iSPIs on the NNMi management server:

- **Windows configuration:**
 - %NnmDataDir%\tmp\HA_nnmhaserver.log
 - %NnmDataDir%\log\haconfigure.log
- **UNIX configuration:**
 - \$NnmDataDir/tmp/HA_nnmhaserver.log
 - \$NnmDataDir/log/haconfigure.log
- **Windows runtime:**
 - **Event Viewer log**
 - %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\ovspmd.log
 - %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\public\postgres.log
 - %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\public\nmsdbmgr.log
 - %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\jbossServer.log
 - %SystemRoot%\Cluster\cluster.log
This is the log file for cluster runtime issues including: adding and removing resources and resource groups; other configuration issues; starting and stopping issues.
- **HP-UX runtime:**
 - /etc/cmcluster/<resource_group>/<resource_group>.cntl.log
This is the log file for the resource group.
 - /var/adm/syslog/syslog.log
 - /var/adm/syslog/OLDSyslog.log
 - \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/ovspmd.log
 - \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/public/postgres.log
 - \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/public/nmsdbmgr.log
 - \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/jbossServer.log
- **Linux runtime for Serviceguard:**
 - /usr/local/cmcluster/conf/<resource_group>/<resource_group>.cntl.log
This is the log file for the resource group.
 - /var/log/cmcluster
This is the log file for cluster issues.
 - /var/log/messages*
These are the log files for resource issues (disk and IP address).
 - \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/ovspmd.log
 - \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/public/postgres.log
 - \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/public/nmsdbmgr.log
 - \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/jbossServer.log

- *Linux or Solaris runtime for VCS:*

Resource	Log File
<code><resource_group>-app</code>	<ul style="list-style-type: none"> • /var/VRTSvcs/log/Application_A.log • \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/ovspmd.log • \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/public/postgres.log • \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/public/nmsdbmgr.log • \$HA_MOUNT_POINT/NNM/dataDir/log/nnm/jbossServer.log • /var/adm/messages*
<code><resource_group>-dg</code> <code><resource_group>-volume</code> <code><resource_group>-mount</code>	<ul style="list-style-type: none"> • /var/VRTSvcs/log/DiskGroup_A.log • /var/VRTSvcs/log/Volume_A.log • /var/VRTSvcs/log/Mount_A.log • /var/adm/messages*
<code><resource_group>-ip</code>	<ul style="list-style-type: none"> • /var/VRTSvcs/log/IP_A.log • /var/adm/messages*

For operating system-specific issues related to the HA resources, review the /var/adm/messages* files. For `<resource_group>-app`, look for messages regarding unable to start process.

Configuring NNMi Advanced for IPv6

You must purchase and install an NNMi Advanced license to use the IPv6 management feature. References to NNMi in this chapter refer to NNMi with an NNMi Advanced license installed.

IPv6 management in NNMi enables the discovery and monitoring of IPv6 addresses, including their interfaces, nodes and subnets. To provide a seamless integration, NNMi extends its IP Address model to include both IPv4 and IPv6 addresses. Whenever possible, NNMi treats all IP Addresses equally; most of the features associated with an IPv4 address are also available for IPv6 addresses. However, there are some exceptions. See the NNMi help for more information about IPv6 information displayed in the NNMi console.

This chapter contains the following topics:

- [Feature Description](#)
- [Prerequisites](#)
- [Licensing](#)
- [Supported Configuration](#)
- [Installing NNMi](#)
- [Activating IPv6 Features](#)
- [Deactivating IPv6 Features](#)

Feature Description

The NNMi IPv6 management feature provides the following:

- IPv6 inventory discovery for IPv6-only and dual-stacked devices
 - IPv6 addresses
 - IPv6 subnets
 - Associations between IPv6 Addresses, Subnets, Interfaces and Nodes
- Native IPv6 SNMP communication for the following:
 - Node discovery

- Interface monitoring
- Trap and inform reception and forwarding
- Automatic selection of IPv4 or IPv6 communication (management address) for dual-stacked devices
 - Configurable preference found in the `nms-jboss.properties` file
- Native ICMPv6 communication for IPv6 Address fault monitoring.
- Seeded device discovery using an IPv6 address or hostname
- Automatic IPv6 device discovery using IPv6 Layer 3 neighbor discovery hints
- Automatic IPv6 device discovery using layer 2 neighbor discovery hints using LLDP (Link Layer Discovery Protocol) IPv6 neighbor information
- Consolidated presentation of IPv4 and IPv6 information
 - Inventory views for nodes, interfaces, addresses, subnets, and associations
 - Layer 2 Neighbor View and Topology Maps for IPv4 and IPv6 devices
 - Layer 3 Neighbor View and Topology Maps for IPv4 and IPv6 devices
 - Incidents, conclusions, root-cause analysis
- NNMi console actions: ping and traceroute for IPv6 addresses and nodes
- NNMi configuration using IPv6 addresses and address ranges
 - Communication configuration
 - Discovery configuration
 - Monitoring configuration
 - Node & Interface Groups
 - Incident configuration
- SDK Web-services support for IPv6 inventory and incidents
- NNM iSPI for Metrics support for IPv6 interfaces

The NNMi IPv6 management feature excludes the following:

- Discovery of IPv6 subnet connections
- Use of IPv6 ping sweep for discovery
- IPv6 Network Path View (Smart Path)
- IPv6 Link Local Address fault monitoring
- Using IPv6 Link Local Addresses as discovery seeds

Prerequisites

Review the *NNMi Deployment Reference*, *NNMi Release Notes*, and *NNMi System and Device Support Matrix* for details on management server specifications and NNMi installation.

To use native IPv6 communication, the NNMi management server must be a dual-stacked system, meaning that it communicates using both IPv4 and IPv6.

IPv6 is not supported on Windows operating systems. See the *NNMi System and Device Support Matrix* for information about the supported operating systems for IPv6. There are other requirements listed below:

- You must enable and configure IPv4 on at least one network interface.
- You must enable IPv6 and have a global unicast address configured on at least one network interface that is connected to the IPv6 network you need to manage.
- You must configure IPv6 routes on the NNMi management server to enable NNMi to communicate with any devices you want NNMi to discover and monitor using IPv6.



You can use an IPv4-only NNMi management server, but doing so will limit NNMi from fully managing IPv4/IPv6 dual-stacked devices. For example, if you use an IPv4-only management server, NNMi cannot discover IPv6-only devices, cannot discover using IPv6 seeds and hints, and cannot monitor for faults on devices having IPv6 addresses.

The DNS server used by the NNMi management server must resolve hostnames to and from IPv6 addresses. For example, it must be able to resolve to and from an AAAA DNS record. That means the DNS server must map a hostname to a 128-bit IPv6 address. If an IPv6-capable DNS server is not available, NNMi will still function correctly; however NNMi does not determine nor display DNS hostnames for nodes using IPv6 addresses.

Licensing

As mentioned earlier, you must purchase and install an NNMi Advanced license to use the IPv6 management feature. See the *NNMi Installation Guide* for information about obtaining and installing your NNMi Advanced license.

The NNMi product includes a temporary Instant-On license password. This is a temporary, but valid NNMi Advanced license. You should obtain and install a permanent license password as soon as possible.

Supported Configuration

See the *NNMi System and Device Support Matrix* for additional information about the supported operating system configurations for NNMi.

Management Server

The following table shows the capabilities of both the IPv4-only and dual-stacked NNMi management server.

Table 20 Management Server Capabilities

Feature/Capability	IPv4-Only	Dual-Stack
IPv4 Communication (SNMP, ICMP)	Supported	Supported
IPv6 Communication (SNMP, ICMPv6)	Not Supported	Supported
Dual-Stack Managed Node	Supported	Supported
Discovery using IPv4 Seed	Supported	Supported
Discovery using IPv6 Seed	Not Supported	Supported
IPv4 Address and Subnet Inventory	Supported	Supported
IPv6 Address and Subnet Inventory	Supported	Supported
Interface Status and Performance using SNMP	Supported	Supported
IPv4 Address Status using ICMP	Supported	Supported
IPv6 Address Status using ICMPv6	Not Supported	Supported
IPv6-only Managed Node	Not Supported	Supported
Discovery using IPv6 Seed	Not Supported	Supported
IPv6 Address and Subnet Inventory	Not Supported	Supported
Interface Status and Performance using SNMP	Not Supported	Supported
IPv6 Address Status using ICMPv6	Not Supported	Supported
IPv4-only Managed Node	Supported	Supported
Node Discovery using IPv4 Seed	Supported	Supported
Node Discovery using IPv4 Seed	Supported	Supported
Interface Status and Performance using SNMP	Supported	Supported
Interface Status and Performance using SNMP	Supported	Supported
IPv4 Address and Subnet Inventory	Supported	Supported

Supported SNMP MIBs for IPv6

NNMi supports the following SNMP MIBs for IPv6:

- RFC 4293 (current IETF standard)
- RFC 2465 (original IETF proposal)
- Cisco IP-MIB

Installing NNMi

During NNMi installation, the installation script includes IPv6 features; however, you must manually enable these IPv6 features. First, you must purchase and apply an NNMi Advanced license to enable the IPv6 features. Then you must manually configure IPv6 to work by editing the `nms-jboss.properties` file. See [Installing NNMi](#) on page 247 for more information.

Activating IPv6 Features

Features requiring IPv6 communication, such as the discovery and of IPv6 only devices and the monitoring of IPv6 address status, require an NNMi management server to have an IPv6 global unicast address configured and operational.

The procedure shown below explains how to enable IPv6 features by doing the following:

- Install an NNMi Advanced license
- Enable the IPv6 Master Switch in the `nms-jboss.properties` file



Review and verify all of the prerequisites described in the preceding section before continuing.

- 1 Use the temporary Instant-on license that comes with NNMi, or install an NNMi Advanced license. See the *NNMi Installation Guide* for information about obtaining and installing NNMi licenses. IPv6 features are not available with the basic NNMi license.
- 2 Edit the `nms-jboss.properties` file. Look in the following location:
 - **UNIX:** `$NNM_PROPS/nms-jboss.properties`
- 3 Locate the text that begins with `# Enable NNMi IPv6 Management`.



NNMi provides a complete description of each property, showing them as comments in the `nms-jboss.properties` file.

- a To enable IPv6 communication in NNMi, un-comment the property:

```
java.net.preferIPv4Stack=false
```



To un-comment a property, remove the `#!` characters from the beginning of a line.

- b To enable overall IPv6 management in NNMi, un-comment the property:

```
com.hp.nnm.enableIPv6Mgmt=true
```

- c Optionally, you can set the SNMP management address preference for dual-stacked managed nodes. Dual-stacked managed nodes are those nodes that can communicate using either IPv4 or IPv6. To do this, un-comment the following line in the `nms-jboss.properties` file:

```
com.hp.nnm.snmp.ipVersionPreference=IPv4
```

Then set the property value to: IPv4, IPv6 or None (no preference). If you do not make **configuration** changes in this area, the SNMP management address preference defaults to IPv4.

- d Save and close the `nms-jboss.properties` file.
- 4 Restart the NNMi management server.
 - a Run the `ovstop` command on the NNMi management server.
 - b Run the `ovstart` command on the NNMi management server.
- 5 Check the NNMi processes using the following command:

```
ovstatus -v ovjboss
```

Successful startup should look something like the following:

```
object manager name: ovjboss
state:                RUNNING
PID:                  <Process ID #>
last message:        Initialization complete.
exit status:          -
additional info:
SERVICE              STATUS
CommunicationModelService  Service is started
CommunicationParametersStatsService  Service is started
EventsCustomExportService  Service is started
ExtensionDeployer        Service is started
IslandSpotterService      Service is started
KeyManager                Service is started
ManagedNodeLicenseManager  Service is started
ModelChangeNotificationAdapter  Service is started
MonitoringSettingsService  Service is started
NMSLogManager             Service is started
NamedPoll                 Service is started
NetworkApplication        Service is started
NmsApa                    Service is started
NmsDisco                  Service is started
NmsEvents                 Service is started
NmsEventsConfiguration    Service is started
NmsExtensionNotificationService  Service is started
NmsModel                  Service is started
NmsWorkManager            Service is started
NnmTrapService            Service is started
RbaConfig                 Service is started
```


RbaManager	Service is started
SpmdjbossStart	Service is started
StagedIcmp	Service is started
StagedSnmp	Service is started
StatePoller	Service is started
TrustManager	Service is started

- 6 After you enable IPv6, NNMi views immediately include the IPv6 inventory for newly discovered nodes. During the next discovery cycle, NNMi views show the IPv6 inventory associated with previously discovered nodes.

To speed things up, select nodes that you know are dual-stack nodes, and then use the **Actions > Configuration Poll** command located in the NNMi console. You can also use the `nmnoderediscover.ovpl` script to add nodes to the NNMi discovery queue. See the `nmnoderediscover.ovpl` reference page, or the UNIX manpage, for more information.

After you enable IPv6 communication on the NNMi management server, NNMi begins monitoring nodes for IPv6 address faults using ICMPv6.

Deactivating IPv6 Features

You can administratively disable IPv6 features using one of the following methods:

- 1 Turn off the IPv6 master switch in the `nms-jboss.properties` file, then restart NNMi.
- 2 Let the NNMi Advanced license expire, or replace it with a basic NNMi license.

The following sections describe NNMi behavior and inventory cleanup after you disable IPv6.

IPv6 Monitoring Following Deactivation

If IPv6 management or IPv6 communication becomes completely disabled, the `StatePoller` service immediately stops monitoring IPv6 addresses with ICMPv6. NNMi sets the IP address state of these addresses to `Not Polled`. If you select an address, then use the **Actions > Monitoring Settings** for this address, NNMi displays `Fault ICMP Polling enabled: false` even though the associated `Monitoring Configuration` rule has the `IP Address Fault Polling enabled`.

IPv6 Inventory Following Deactivation

Once NNMi completely discovers your IPv6 inventory, you can enable NNMi to clean it up automatically in the following scenarios:

- You turned on the master IPv6 switch, then turned it off and restarted NNMi.
 NNMi does not immediately remove the IPv6 inventory. NNMi removes the IPv6 inventory for SNMP nodes during the next discovery cycle. NNMi does not remove non-SNMP IPv6 nodes. You need to manually delete IPv6 nodes from the NNMi inventory.

- Your NNMi Advanced license expired or someone removed the license. NNMi begins using the NNMi basic license, and the basic license has enough capacity to continue managing all of the discovered nodes.

NNMi immediately removes all of the non-SNMP IPv6 nodes from its inventory. NNMi rediscovers all of the SNMP nodes and removes all of the IPv6 data.

- Your NNMi Advanced license expired or someone removed the license. NNMi begins using the NNMi basic license, and the basic license does not have enough capacity to continue managing all of the discovered nodes. NNMi immediately removes all non-SNMP IPv6 nodes. The `Licensing` service marks the SNMP nodes that exceed the licensed inventory capacity with an `unmanaged` state. NNMi immediately removes IPv6 data from the managed SNMP nodes.

For the unmanaged SNMP nodes, complete these steps:

- a Install additional license capacity.
 - b Use the **Actions > Management Mode > Manage** command located in the NNMi console to change the management mode for the nodes marked as `unmanaged` by the `Licensing` service. You can use the `nnmmanagementmode.ovpl` script to manage these nodes as well. See the `nnmmanagementmode.ovpl` reference page, or the UNIX manpage, for more information.
 - c Use the **Actions > Configuration Poll** command located in the NNMi console to enable NNMi to discover them. You can use the `nnmnoderediscover.ovpl` script to discover these nodes as well. See the `nnmnoderediscover.ovpl` reference page, or the UNIX manpage, for more information.
- Your NNMi Advanced license expired or someone removed the license; You neglected to install an NNMi basic license.

NNMi immediately removes all non-SNMP IPv6 nodes and automatically unmanages the remaining nodes. To remedy this situation, complete these steps:

- a Install a valid license.
- b Use the **Actions > Management Mode > Manage** command located in the NNMi console to change the management mode for the nodes marked as `unmanaged` by the `Licensing` service. You can use the `nnmmanagementmode.ovpl` script to manage these nodes as well. See the `nnmmanagementmode.ovpl` reference page, or the UNIX manpage, for more information.
- c Use the **Actions > Configuration Poll** command located in the NNMi console to enable NNMi to discover the nodes you changed from `unmanaged` to `managed`. You can use the `nnmnoderediscover.ovpl` script to discover these nodes as well. See the `nnmnoderediscover.ovpl` reference page, or the UNIX manpage, for more information.
- d To create an IPv6 list, then remove the IPv6 inventory, use the **Actions > Configuration Poll** command to obtain configuration information from each managed node.

Known Issues When Cleaning Up IPv6 Inventory

You could experience leftover IPv6 inventory in the following situation: Suppose that NNMi successfully uses SNMP to manage an IPv6 node, then the node becomes inaccessible before the next discovery. Due to the design of the existing discovery system, the discovery process cannot update a node that loses its ability to communicate using SNMP. To remove these remaining nodes, you need to fix the

communication problem, then use the **Actions > Configuration Poll** command located in the NNMi console to obtain configuration information from these nodes. For native IPv6 nodes, delete the node directly from the NNMi console.

Maintaining NNMi

This section contains the following chapters:

- [NNMi Backup and Restore Tools](#)
- [Maintaining NNMi](#)
- [Configuring the Telnet and SSH Protocols for Use by NNMi](#)
- [Changing the NNMi Management Server](#)

NNMi Backup and Restore Tools

A good backup and restore strategy is key to ensuring the uninterrupted operations of any business. HP Network Node Manager i Software is an important asset for network operations and should be backed up regularly.

The two types of critical data related to an NNMi installation are as follows:

- Files in the file system
- Data in the relational database (embedded or external)

This chapter explains the tools that NNMi provides for backing up and restoring important NNMi files and data.

This chapter contains the following topics:

- [Backup and Restore Commands](#)
- [Backing up NNMi Data](#)
- [Restoring NNMi Data](#)
- [Backup and Restore Strategies](#)
- [Backing up and Restoring the Embedded Database Only](#)

Backup and Restore Commands

NNMi provides the following scripts for backing up and restoring NNMi data:

- `nnmbackup.ovpl`—Backs up all necessary file system data (including configuration information) and any data stored in the NNMi embedded database.
- `nnmrestore.ovpl`—Restores a backup that was created by using the `nnmbackup.ovpl` script.
- `nnmbackupembdb.ovpl`—Creates a complete backup of the NNMi embedded database (but not the file system data) while NNMi is running.
- `nnmrestoreembdb.ovpl`—Restores a backup that was created by using the `nnmbackupembdb.ovpl` script.

- `nnmresetembdb.ovpl`—Drops the NNMi embedded database tables. Run the `ovstart` command to recreated the tables.

For command syntax, see the appropriate reference page, or the UNIX manpage.

Backing up NNMi Data

The NNMi backup command (`nnmbackup.ovpl`) copies key NNMi file system data and some or all of the tables in the NNMi Postgres database to the specified target directory. The NNMi backup command can create a tar archive of the backup data, or you can compress the backup files using your own tools. You can then use any appropriate tool to save a copy of the backup.



If your NNMi implementation uses Oracle for the main NNMi database, the NNMi backup and restore commands work with the NNMi file system data only. External database maintenance should be handled as part of the existing database backup and restore procedures.

The back up and restore data may or may not include data from any NNM iSPIs installed in your network environment. Check the documentation that came with each NNM iSPI for details.



Any software that locks files (for example, anti-virus or system backup software), can interrupt NNMi access to the NNMi database, which can cause problems. For the NNMi Postgres database, configure these applications to exclude the NNMi database directory (`%NNM_DB%` and Windows, `$NNM_DB` on UNIX). Use `nnmbackup.ovpl` to back up the NNMi database regularly.

Backup Type

The NNMi backup command supports two types of backups:

- Online backups occur while NNMi is running. NNMi ensures that the database tables are synchronized in the backed up data. Operators can be actively using the NNMi console and other processes can be interacting with the NNMi database during an online backup. With an online backup, you can back up all NNMi data or only some of the data according to function, as described in [Backup Scope](#). For the embedded NNMi database, the `nmsdbmgr` service must be running. For an external database, the backup includes NNMi file system data only and no NNMi processes must be running.
- Offline backups occur while NNMi is completely stopped. With an offline backup, the backup scope applies to the file system files only. An offline backup always includes the complete NNMi database regardless of the backup scope. For the embedded NNMi database, the backup copies the Postgres database files. For an external database, the backup includes NNMi file system data only.

Backup Scope

The NNMi backup command provides several scopes that define how much NNMi is backed up.

Configuration scope

The configuration scope (`-scope config`) loosely aligns to the information in the **Configuration** workspace of the NNMi console.

The configuration scope includes the following data:

- For online backups, only those embedded database tables that store NNMi configuration information.
- For offline backups, the entire embedded database.
- For all backups, the NNMi configuration information in the file system as listed in [Table 21](#).

Topology scope

The topology scope (`-scope topology`) loosely aligns to the information in the **Inventory** workspace of the NNMi console. Because the network topology is dependent on the configuration that was used for discovering that topology, the topology scope includes the configuration scope.

The topology scope includes the following data:

- For online backups, only those embedded database tables that store NNMi configuration and network topology information.
- For offline backups, the entire embedded database.
- For all backups, the NNMi configuration information in the file system as listed in [Table 21](#). Currently, there are no file system files associated with the topology scope.

Event scope

The event scope (`-scope event`) loosely aligns to the information in the **Incident Browsing** workspace of the NNMi console. Because events are dependent on the network topology related to those events, the event scope includes the configuration and topology scopes.

The event scope includes the following data:

- For online backups, only those embedded database tables that store NNMi configuration, network topology, and event information.
- For offline backups, the entire embedded database.
- For all backups, the NNMi configuration information in the file system as listed in [Table 21](#) and the NNMi event information as listed in [Table 22](#).

All scope

The complete backup (`-scope all`) includes all important NNMi files and the complete embedded database.

Table 21 Configuration Scope Files and Directories

Directory or File name	Description
<code>\$NnmInstallDir/conf</code> (Windows only)	Configuration information
<code>\$NnmInstallDir/misc/nnm</code>	Miscellaneous configuration information
<code>\$NnmInstallDir/misc/nms/lic</code>	Miscellaneous license information
<code>\$NnmInstallDir/newconfig</code>	Installation configuration staging area

Table 21 Configuration Scope Files and Directories (cont'd)

Directory or File name	Description
\$NnmInstallDir/nonOV/jboss/nms/server/nms/conf	jboss configuration
\$NnmInstallDir/nonOV/jboss/nms/server/nms/deploy	jboss deployment directory
\$NnmDataDir/conf	Configuration that might be shared by other HP products
\$NnmDataDir/conf/nnm/props	Local NNMi configuration properties files
<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> <drive>:\ProgramData\Hewlett-Packard\HPOvLIC\data\LicFile.txt • <i>Windows Server 2003:</i> <drive>:\Program Files (x86)\Common Files\Hewlett-Packard\HPOvLic\data\LicFile.txt • <i>UNIX:</i> \$NnmDataDir/HPOvLIC/LicFile.txt 	License information
\$NnmDataDir/NNMVersionInfo	NNMi version information file
\$NnmDataDir/shared/nnm/user-snmp-mibs	Shared user-added SNMP MIB information
\$NnmDataDir/shared/nnm/actions	Shared lifecycle transition actions
\$NnmDataDir/shared/nnm/certificates	Shared NNMi SSL certificates
\$NnmDataDir/shared/nnm/conf	Shared NNMi configuration information
\$NnmDataDir/shared/nnm/conf/licensing	Shared NNMi license configuration information
\$NnmDataDir/shared/nnm/lrf	Shared NNMi component registration files
\$NnmDataDir/shared/nnm/conf/props	Shared NNMi configuration properties files
\$NnmDataDir/shared/nnm/www/htdocs/images	Shared background images for NNMi node group maps

In this context, files in the shared directories are those shared with another NNMi management server in an NNMi application failover or high availability environment.

Table 22 Event Scope Files and Directories

Directory or File name	Description
\$NnmDataDir/log/nnm/signin.0.0.log	NNMi console sign-in log

Restoring NNMi Data

The NNMi restore script (`nnmrestore.ovpl`) places the backup data on the NNMi management server. The type and scope of the backup determines what NNMi is able to restore.

- ▶ If you use the `nnmrestore.ovpl` script to place database records on a second NNMi management server, both NNMi management servers must have the same type of operating system and NNMi version and patch level.

Placing the backup data from one NNMi management server onto a second NNMi management server means that both servers have the same database UUID. After you restore NNMi on the second NNMi management server, uninstall NNMi from the original NNMi management server.

- To restore an online backup, NNMi copies the file system data to the correct locations and overwrites the contents of the database tables that were included in the backup. Objects that have been deleted since the backup are restored, and objects that have been created since the backup are deleted. Additionally, any objects that were changed after the backup was taken revert to their state at the time of the backup. For the embedded NNMi database, the `nmsdbmgr` service must be running. For an external database, the restore includes NNMi file system data only and no NNMi processes must be running.
- To restore an offline backup, NNMi overwrites the Postgres files in the file system, completely replacing the database files with the contents of the backup. For an external database, the backup includes NNMi file system data only.

With the `-force` option, the `nnmrestore.ovpl` command stops all NNMi processes, starts the `nmsdbmgr` service (if restoring from an online backup of the NNMi embedded database), restores the data, and then restarts all NNMi processes.

If the provided source is a tar file, the NNMi restore command extracts the tar file to a temporary folder in the current working directory. In this case, either ensure that the current working directory has adequate storage to support the temporary folder, or extract the archive before running the restore command.

- ▶ Because the database schema might change from one version of NNMi to the next, data backups cannot be shared across versions of NNMi.

Same System Restore

You can use the backup and restore commands on a single system for data recovery. The following items must not have changed between the time of the backup and time of the restore:

- NNMi version (including any patches)
- Operating system type and version
- Character set (language)
- Hostname
- Domain

Different System Restore

You can use the backup and restore commands to transfer data from one NNMi management server to another. The intended uses of different system restoration include recovering from system failure and transferring NNMi to a different system during an operating system upgrade.

Best practice

Because the NNMi UUID is copied to the target system during the database restore, both source and target systems now appear to be running the same instance of NNMi. Uninstall NNMi from the source system.



To create multiple functional NNMi management servers with similar configurations, such as while deploying global network management, use the `nnmconfigexport.ovpl` and `nnmconfigimport.ovpl` commands.

For a different system restore, the following items must be identical on both systems:

- NNMi version (including any patches)
- Operating system type and version
- Character set (language)

The following items can differ between the two systems:

- Hostname
- Domain

For a different system restore, the `nnmrestore.ovpl` command does not copy the license information to the new system. Obtain and apply a new license for the new NNMi management server. For more information, see the licensing documentation.

Backup and Restore Strategies

Back up All Data Periodically

Your disaster recovery plan should include a regularly scheduled complete backup of all NNMi data. You do not need to shut down NNMi to create this backup. If you incorporate the backup into a script, use the `-force` option to ensure that NNMi is on the correct state before the backup begins. For example:

```
nnmbackup.ovpl -force -type online -scope all -archive \  
-target nmi_backups\periodic
```

If you need to recover your NNMi data after a hardware failure, follow these steps:

- 1 Rebuild or acquire new hardware.
- 2 Install NNMi to the same version and patch level as were in place for the backup.
- 3 Restore the NNMi data:
 - If the recovery NNMi management server meets the requirements listed in [Same System Restore](#) on page 259, run a command similar to the following example:

```
nnmrestore.ovpl -force -lic \  
-source nmi_backups\periodic\newest_backup
```

- If the recovery NNMi management server does not qualify for a same-system restore but meets the requirements listed in [Different System Restore](#) on page 260, run a command similar to the following example:

```
nmrestore.ovpl -force \  
-source nmi_backups\periodic\newest_backup
```

Update the licensing as needed.

Back up Data Before Changing the Configuration

Perform scoped backups (as described in [Backup Scope](#) on page 257) as needed before beginning configuration changes. In this way, if your configuration changes do not have the expected effect, you will be able to revert to a known working configuration. For example:

```
nmbackup.ovpl -type online -scope config \  
-target nmi_backups\config
```

To restore this backup to the same NNMi management server, stop all NNMi processes, and then run a command similar to the following example:

```
nmrestore.ovpl -force -source nmi_backups\config\newest_backup
```

Back up Data Before Upgrading NNMi or the Operating System

Before making major system changes (including upgrading NNMi or the operating system), perform a complete backup of all NNMi data. 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:

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

If NNMi does not run correctly after the system change, roll back the change or set up a different NNMi management server and ensure that the requirements listed in [Different System Restore](#) on page 260 are met. Then run a command similar to the following example:

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

Restore File System Files Only

To overwrite NNMi files without affecting the database tables, run a command similar to the following example:

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

The command is useful when the NNMi implementation uses Oracle for the main NNMi database.

Backing up and Restoring the Embedded Database Only

NNMi provides the `nnmbackupembdb.ovpl` and `nnmrestoreembdb.ovpl` commands to back up and restore the NNMi embedded database only. This functionality is useful for creating a snapshot of the data as you experiment with NNMi configuration settings. The `nnmbackupembdb.ovpl` and `nnmrestoreembdb.ovpl` commands perform online backups only. At a minimum, the `nmsdbmgr` service must be running.

Best practice

Run the `nnmresetembdb.ovpl` command before restoring data to the embedded database. This command ensures that the database does not contain any errors, thereby eliminating the possibility of encountering database constraint violations. For information about running the embedded database reset command, see the `nnmresetembdb.ovpl` reference page, or the UNIX manpage.

Maintaining NNMi

After you have your NNMi management server functioning, there are maintenance tasks you can perform to optimize several of the NNMi features.

This chapter contains the following topics:

- [Administering a Custom Poller Collection Export](#) on page 263
- [Administering Incident Actions](#) on page 265
- [Modifying NNMi Normalization Properties](#) on page 267
- [Configure HTTPS-Only Communication with the NNMi Console](#) on page 269

Administering a Custom Poller Collection Export

The **Custom Poller feature** enables you to take a proactive approach to network management by using SNMP MIB expressions to specify additional information that NNMi should poll. A **Custom Poller collection** defines the information you want to gather (poll) as well as how NNMi reacts to the gathered data. See *Create a Custom Poller Collection* and *Configure Custom Polling* in the NNMi help for more detailed information.

The CustomPoller feature relies on you to remove files from the export directory as you process them. Do not use the exported files for long term storage; if they consume more than the configured maximum disk space, NNMi removes the older files and creates new ones. Unless you process these files and store them in a different location, you will lose them.

Changing the Custom Poller Collections Export Directory

NNMi writes the data from the collections you export into the following directory:

- **Windows:** %NNM_DATA%\shared\nnm\databases\custompoller\export
- **UNIX:** \$NNM_DATA/shared/nnm/databases/custompoller/export

To change the directory that NNMI writes its custom poller files into, follow these steps:

1 Edit the following file:

- *Windows*: %NNM_PROPS%\nms-custompoller.properties
- *UNIX*: \$NNM_PROPS/nms-custompoller.properties

2 Look for the `exportdir` entry, which is similar to the following line:

```
#!com.hp.nnm.custompoller.exportdir=<base directory to export custom poller metrics>
```

To configure NNMI to write Custom Poller collection information into the `C:\CustomPoller` directory, change the line as follows:

```
com.hp.nnm.custompoller.exportdir=C:\CustomPoller
```

3 Restart the NNMI management server.

- a Run the `ovstop` command on the NNMI management server.
- b Run the `ovstart` command on the NNMI management server.

Changing the Maximum Amount of Disk Space for Custom Poller Collections Export

To change the maximum amount of disk space that NNMI uses when exporting data to `collection_name.csv` files, follow these steps:

1 Edit the following file:

- *Windows*: %NNM_PROPS%\nms-custompoller.properties
- *UNIX*: \$NNM_PROPS/nms-custompoller.properties

2 Look for the `maxdiskspace` entry, which is similar to the following line:

```
#!com.hp.nnm.custompoller.maxdiskspace=1000
```

To configure NNMI to reserve up to 2000 MB (2 GB) of storage space for each `collection_name.csv` file, change the line as follows:

```
#!com.hp.nnm.custompoller.maxdiskspace=2000
```

3 Restart the NNMI management server.

- a Run the `ovstop` command on the NNMI management server.
- b Run the `ovstart` command on the NNMI management server.

Changing the Custom Poller Metric Accumulation Interval

NNMI sets the length of time, in minutes, that it accumulates Custom Poller Collection metrics before it writes data into a file.

To change the custom poller metric accumulation interval, follow these steps:

1 Edit the following file:

- *Windows*: %NNM_PROPS%\nms-custompoller.properties
- *UNIX*: \$NNM_PROPS/nms-custompoller.properties

- 2 Look for a line that resembles the following:

```
#!com.hp.nnm.custompoller.accumulationinterval=5
```

To configure NNMi to collect metrics for ten minutes instead of the default value of five minutes, change the line as follows:

```
com.hp.nnm.custompoller.accumulationinterval=10
```

- 3 Restart the NNMi management server.
 - a Run the `ovstop` command on the NNMi management server.
 - b Run the `ovstart` command on the NNMi management server.

Administering Incident Actions

You can configure actions to automatically run at any point in the incident lifecycle. For example, you might want to configure an action to occur when an incident of the type you are configuring is generated. See *Configure an Action for an Incident* in the NNMi Help for more information.

To adjust action parameters, follow the steps shown in the following sections.

Setting the Number of Simultaneous Actions



Increasing the number of simultaneous actions on a Solaris NNMi management server causes NNMi performance degradation.

To modify the number of simultaneous actions that NNMi can run, follow these steps:

- 1 Edit the following file:
 - *Windows:* %NNM_PROPS%\shared\nnmaction.properties
 - *UNIX:* \$NNM_PROPS/shared/nnmaction.properties
- 2 Look for a line that resembles the following:

```
#!com.hp.ov.nms.events.action.numProcess=10
```

To configure NNMi to enable 20 simultaneous actions instead of the default value, change the line as follows:

```
com.hp.ov.nms.events.action.numProcess=20
```



Make sure to remove the `#!` characters located at the beginning of the line.

- 3 Restart the NNMi management server.
 - a Run the `ovstop` command on the NNMi management server.
 - b Run the `ovstart` command on the NNMi management server.

Setting the Number of Threads for Jython Actions

To modify the number of threads the action server uses to run jython scripts, follow these steps:

- 1 Edit the following file:

- *Windows*: %NNM_PROPS%\shared\nmaction.properties
- *UNIX*: \$NNM_PROPS/shared/nmaction.properties

- 2 Look for a line that resembles the following:

```
#!com.hp.ov.nms.events.action.numJythonThreads=10
```

To configure NNMi to enable 20 threads for running jython scripts instead of the default value, change the line as follows:

```
com.hp.ov.nms.events.action.numJythonThreads=20
```



Make sure to remove the **#!** characters located at the beginning of the line.

- 3 Restart the NNMi management server.

- a Run the **ovstop** command on the NNMi management server.
- b Run the **ovstart** command on the NNMi management server.

Setting the Action Server Name Parameter

To modify the user name that runs the action server on a Windows NNMi management server, change the `LogOn` property of the HP NNM Action Server service.

To modify the user name that runs the action server (applicable to HP-UX, Solaris, and Linux NNMi management servers), follow these steps:

- 1 Edit the following file:

```
$NNM_PROPS/shared/nmaction.properties
```

- 2 Look for a line the resembles the following:

```
#!com.hp.ov.nms.events.action.userName=bin
```

To configure NNMi to have *system* run the action server instead of the default value, change the line as follows:

```
com.hp.ov.nms.events.action.userName=system
```



Make sure to remove the **#!** characters located at the beginning of the line.

- 3 Save your changes.

Changing the Action Server Queue Size

For actions that use a long action command string at a high execution rate, such as responding to a trap storm, the action server can use up a lot of memory. To provide better action server performance, HP places limits on the memory size that the action server can grow to.



For Solaris NNMi management servers, if the NNMi health information shows that action queue sizes are growing, reduce the maximum memory size to improve performance.

To modify these limits, follow these steps:

- 1 Edit the following file:
 - `%NNM_PROPS%\shared\nnmaction.properties`
 - `$NNM_PROPS/shared/nnmaction.properties`
- 2 Look for two lines that resemble the following:

```
com.hp.ov.nms.events.action.jvmargs.minMemsize=-Xms6m
com.hp.ov.nms.events.action.jvmargs.maxMemsize=-Xmx30m
```
- 3 The above parameters show the minimum memory size set to 6MB and the maximum set to 30MB. Adjust these parameters to meet your needs.
- 4 Save your changes.
- 5 Restart the NNMi management server.
 - a Run the `ovstop` command on the NNMi management server.
 - b Run the `ovstart` command on the NNMi management server.

Modifying NNMi Normalization Properties

NNMi stores both hostnames and node names in case-sensitive form. This means that all searches, sorts, and filters that the NNMi console provides return case-sensitive results. If the DNS servers you use return a variety of case-preserving node names and hostnames, including all uppercase, all lowercase, and a mixture of uppercase and lowercase, this can cause less-than-optimal results.

You can change several NNMi normalization properties to meet your specific needs. A good practice is to make these changes prior to running the NNMi Quick Start Configuration Wizard or seeding NNMi for its initial discovery.

HP recommends that you adjust the settings in this section during deployment, but prior to running the initial discovery.

If you run an initial discovery, then decide to change the normalization properties later, you can run the `nnmnodediscover.ovpl -all` script to initiate a full discovery. See the `nnmnodediscover.ovpl` reference page, or the UNIX manpage, for more information.

You can change the following properties:

- Normalize discovered node names to UPPERCASE, LOWERCASE, or OFF.
- Normalize discovered hostnames to UPPERCASE, LOWERCASE, or OFF.


To change normalization properties follow these steps:

- 1 Edit the following file:
 - *Windows*: %NNM_PROPS%\nrm-topology.properties
 - *UNIX*: \$NNM_PROPS/nrm-topology.properties
- 2 To configure NNMi to normalize discovered names, look for a line the resembles the following:

```
#!com.hp.ov.nms.topo.NAME_NORMALIZATION=OFF
```

 - a Un-comment the property:

```
com.hp.ov.nms.topo.NAME_NORMALIZATION=OFF
```

 To un-comment a property, remove the #! characters from the beginning of a line.
 - b Change OFF to LOWERCASE or UPPERCASE.
 - c Save your changes.
- 3 To configure NNMi to normalize discovered hostnames, look for a line the resembles the following:

```
#!com.hp.ov.nms.topo.NAME_NORMALIZATION=OFF
```

 - a Un-comment the property:

```
com.hp.ov.nms.topo.HOSTNAME_NORMALIZATION=OFF
```
 - b Change OFF to LOWERCASE or UPPERCASE.
 - c Save your changes.
- 4 Restart the NNMi management server.
 - a Run the **ovstop** command on the NNMi management server.
 - b Run the **ovstart** command on the NNMi management server.

Changing Normalization Properties Following an Initial Discovery

Changing normalization properties following an initial discovery causes NNMi to be inconsistent with the property changes until the next discovery. To remedy this, run the **nnmnode Rediscover.ovpl -all** script to initiate a full discovery after changing NNMi normalization properties.

After NNMi completes a full discovery, the behaviors shown below should return to normal. These examples are not exhaustive, and are intended to provide a few examples of things to consider when changing NNMi normalization properties.

Configure HTTPS-Only Communication with the NNMi Console

The most effective method of preventing HTTP access to the NNMi console is to place the NNMi management server behind a firewall that permits only HTTPS access to the protected systems.

Firewall configuration to prevent HTTP access can cause problems for integrations that use web services to communicate with NNMi and only support HTTP. See the documentation for the integrating product to find out if it supports HTTPS.

For a less secure approach, redirect NNMi console access requests from the HTTP port to the HTTPS port by completing the following steps:

1 Edit the following file:

- *Windows*: %NNM_PROPS\nms-ui.properties
- *UNIX*: \$NNM_PROPS/nms-ui.properties

2 Search for the string `https` to locate the text block containing the following line:

```
#! com.hp.ov.nms.ui.https.only=false
```

3 Uncomment and edit the following line to read as follows:

```
com.hp.ov.nms.ui.https.only=true
```

4 Restart `ovjboss`:

```
ovstop ovjboss
ovstart ovjboss
```



Setting this property to redirect HTTP requests to HTTPS for the NNMi console can cause problems with some applications that cross-launch back to NNMi. If you experience these problems, disable this HTTPS redirect.

NNMi Self Monitoring

NNMi performs self-monitoring checks, including memory, CPU, and disk resources. NNMi generates an incident after the NNMi management server becomes low on resources or detects a serious condition.

To view NNMi health information, use one of the following methods:

- From the NNMi console, click **View > System Information**; then click the **Health** tab.
- For a detailed self-monitoring report, select **Tools > NNMi System Health Report**
- Run the `nnmhealth.ovpl` script.

NNMi displays a status message at the bottom of the NNMi console and on the top of forms after an NNMi detects a self-monitoring health exception. You can disable this warning message by completing the following steps:

1 Edit the following file:

- *Windows*: %NNM_PROPS\nms-ui.properties
- *UNIX*: \$NNM_PROPS/nms-ui.properties

- 2 Locate the text block containing the following line:
`#!com.hp.nms.ui.health.disablewarning=false`
- 3 Uncomment and edit the following line to read as follows:
`com.hp.nms.ui.health.disablewarning==true`
- 4 Restart ovjboss:
 - a **ovstop ovjboss**
 - b **ovstart ovjboss**

Configuring the Telnet and SSH Protocols for Use by NNMi

The **Actions > Telnet... (from client)** menu item invokes the telnet command to the selected node (from the web browser in which the NNMi console is currently running). By default, neither Microsoft Internet Explorer nor Mozilla Firefox defines the telnet command, so using the **Telnet** menu item produces an error message. You can configure the telnet, secure shell (SSH), or both protocols for each NNMi user (on a per-system basis), and you can change the NNMi console menu items.

This chapter contains the following topics:

- [Disable the Telnet Menu Item](#) on page 271
- [Configure a New Menu Item that uses the ssh Protocol](#) on page 272
- [Configure a Telnet or SSH Client for the Browser on Windows](#) on page 272
- [Configure Firefox to use Telnet or SSH on Linux](#) on page 279
- [Example Files for Changing the Windows Registry](#) on page 280

Disable the Telnet Menu Item

If the NNMi users in your deployment environment do not require telnet connections from the NNMi console, you can disable the **Telnet** menu item to remove it from the NNMi console.

For information about creating a new menu item for SSH connections to managed devices, see [Configure a New Menu Item that uses the ssh Protocol](#) on page 272.

Disabling the **Telnet** menu item in the NNMi console applies to all users who sign in to the NNMi console on this NNMi management server. To disable the **Telnet** menu item, follow these steps:

- 1 In the **Configuration** workspace, click **User Interface Configuration**.
- 2 On the **Menu Items** tab, select the **Telnet... (from Client)** row, and then click **Open**



- 3 On the **Menu Item** form, clear the **Enabled** check box, and then set the **Author** field to an appropriate value.
Changing the author value ensures that this menu item remains disabled when you upgrade NNMi.
- 4 Save and close the form.

For more information, see *Control the Actions Menu* in the NNMi help.

Configure a New Menu Item that uses the ssh Protocol

Configuring a new menu item for secure shell access in the NNMi console applies to all users who sign in to the NNMi console on this NNMi management server.

Create a new menu item on the Actions menu that includes the following menu item contexts:

- For the *node* object type, set the full URL to: `ssh://${hostname}`
- For the *IP address* object type, set the full URL to: `ssh://${value}`

For more information, see *Control the Actions Menu* in the NNMi help.

Configure a Telnet or SSH Client for the Browser on Windows

Configure the operating-system provided telnet command for an NNMi user's web browser. This procedure must be done for each computer and web browser from which an NNMi user needs to run the **Actions > Telnet... (from Client)** menu item.

To complete any of the procedures in this section, you must have administrative privileges on the computer. The specific steps depend on the version (32-bit or 64-bit) of the browser and the operating system.

To determine the version of Internet Explorer, click **Help > About Internet Explorer**. If the version information does not include the text **64-bit Edition**, this Internet Explorer is 32-bit.

Firefox is only available in a 32-bit version.

Table 23 identifies the procedure to use for each browser and operating system combination.

Table 23 Matrix of Telnet and SSH Configuration Procedures on Windows

Web Browser	Windows Operating System Architecture	Applicable Procedures
Internet Explorer 32-bit	32-bit	<ul style="list-style-type: none"> Windows Operating System-Provided Telnet Client on page 274 Third-Party Telnet Client (Standard Windows) on page 275 Third-Party SSH Client (Standard Windows and Windows on Windows) on page 277
	64-bit Windows 7	<ul style="list-style-type: none"> Third-Party Telnet Client (Standard Windows) on page 275 Third-Party SSH Client (Standard Windows and Windows on Windows) on page 277
	64-bit other than Windows 7	<ul style="list-style-type: none"> Third-Party Telnet Client (Windows on Windows) on page 276 Third-Party SSH Client (Standard Windows and Windows on Windows) on page 277
Internet Explorer 64-bit	64-bit	<ul style="list-style-type: none"> Windows Operating System-Provided Telnet Client on page 274 Third-Party Telnet Client (Standard Windows) on page 275 Third-Party SSH Client (Standard Windows and Windows on Windows) on page 277
Firefox	32-bit	<ul style="list-style-type: none"> Windows Operating System-Provided Telnet Client on page 274 Third-Party Telnet Client (Standard Windows) on page 275 Third-Party SSH Client (Standard Windows and Windows on Windows) on page 277
	64-bit Windows 7	<ul style="list-style-type: none"> Third-Party Telnet Client (Standard Windows) on page 275 Third-Party SSH Client (Standard Windows and Windows on Windows) on page 277
	64-bit other than Windows 7	<ul style="list-style-type: none"> Third-Party Telnet Client (Windows on Windows) on page 276 Third-Party SSH Client (Standard Windows and Windows on Windows) on page 277



Many of the tasks in this section involve editing the Windows registry. Instead of editing the registry directly, you can create a .reg file that each user can run on their system. For example .reg files, see [Example Files for Changing the Windows Registry](#) on page 280.

For more information about the tasks described in this section, see the following Microsoft articles:

- Installing the Microsoft-provided telnet client:
<http://technet.microsoft.com/en-us/library/cc771275%28WS.10%29.aspx>
- Introduction to the Windows registry:
<http://support.microsoft.com/kb/256986>
- Backing up and restoring the Windows registry:
<http://support.microsoft.com/kb/322756>

Windows Operating System-Provided Telnet Client

This procedure applies to the following cases:

- 32-bit Internet Explorer on a 32-bit operating system
- 32-bit Firefox on a 32-bit operating system
- 64-bit Internet Explorer on a 64-bit operating system

To configure the operating system-provided telnet client for use by a web browser, follow these steps:

- 1 (Microsoft Windows 7, Microsoft Vista, or Microsoft Windows Server 2008 only) Install the operating system telnet client on the computer by following the steps appropriate to the operating system.

Windows 7 or Vista:

- a In the Control Panel, click **Programs**, and then click **Programs and Features**.
- b Under Tasks, click **Turn Windows features on or off**.
- c In the Windows Features dialog box, select the **Telnet Client** check box, and then click **OK**.

Windows Server 2008:

- a In the Server Manager, under Features Summary, click **Add Features**.
- b In the Add Features Wizard, select the **Telnet Client** check box, click **Next**, and then click **Install**.

- 2 (Internet Explorer only) Enable Internet Explorer to use the telnet protocol.

- a Back up the Windows registry.
- b Use the Windows registry editor to add the [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL] key with the following values:

Name	Type	Data
iexplore.exe	REG_DWORD	0

- 3 Set file association for the URL:Telnet Protocol file type.
 - a Back up the Windows registry.
 - b Use the Windows registry editor to modify the [HKEY_CLASSES_ROOT\
 telnet\shell\open\command] key with the following value:

Name	Type	Data
(default)	REG_SZ	rundll32.exe url.dll,TelnetProtocolHandler %l

%l (with a lowercase L) is the argument passed to telnet, usually an IP address or the fully-qualified domain name of a node.



For tighter control, you can encode the paths to the binaries in the key (as a single line). For example:

```
"C:\Windows\system32\rundll32.exe"
"C:\Windows\system32\url.dll",TelnetProtocolHandler %l
```

- 4 Restart the web browser, and then, in the browser address bar, enter the telnet command:

```
telnet://<node>
```

<node> is the IP address or fully-qualified domain name of a node that runs the telnet server.

If you are prompted with a security warning, allow the action.

In Firefox, select the **Remember my choice for telnet links** check box.

Third-Party Telnet Client (Standard Windows)

This procedure applies to the following cases:

- 32-bit Internet Explorer on a 32-bit operating system
- 32-bit Internet Explorer on a 64-bit Windows 7 operating system
- 32-bit Firefox on a 32-bit operating system
- 64-bit Internet Explorer on a 64-bit operating system

To configure a third-party telnet client for use by a web browser, follow these steps:

- 1 Obtain and install a third-party telnet client.

This procedure gives examples for the PuTTY client installed to C:\Program Files\PuTTY\putty.exe. The PuTTY client is available from <http://www.putty.org>.

- 2 (Internet Explorer only) Enable Internet Explorer to use the telnet protocol.

- a Back up the Windows registry.
- b Use the Windows registry editor to add the [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL] key with the following values:

Name	Type	Data
iexplore.exe	REG_DWORD	0

- 3 Set file association for the URL:Telnet Protocol file type.
 - a Back up the Windows registry.
 - b Use the Windows registry editor to modify the [HKEY_CLASSES_ROOT\telnet\shell\open\command] key with the following value:

Name	Type	Data
(default)	REG_SZ	"C:\Program Files\PuTTY\putty.exe" %l

%l (with a lowercase L) is the argument passed to telnet, usually an IP address or the fully-qualified domain name of a node.



In a .reg file, escape each quotation mark (") and backslash (\) character with a backslash (\) character.

- 4 Restart the web browser, and then, in the browser address bar, enter the telnet command:

```
telnet://<node>
```

<node> is the IP address or fully-qualified domain name of a node that runs the telnet server.

If you are prompted with a security warning, allow the action.

In Firefox, select the **Remember my choice for telnet links** check box.

Third-Party Telnet Client (Windows on Windows)

This procedure applies to the following cases:

- 32-bit Internet Explorer on a 64-bit operating system (other than Windows 7)
- 32-bit Firefox on a 64-bit operating system

To configure a third-party telnet client for use by a web browser, follow these steps:

- 1 Obtain and install a third-party telnet client.

This procedure gives examples for the PuTTY client installed to C:\Program Files\PuTTY\putty.exe. The PuTTY client is available from <http://www.putty.org>.

- 2 (Internet Explorer only) Enable Internet Explorer to use the telnet protocol.

- a Back up the Windows registry.
- b Use the Windows registry editor to add the [HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL] key with the following values:

Name	Type	Data
iexplore.exe	REG_DWORD	0

- 3 Set file association for the URL:Telnet Protocol file type.
 - a Back up the Windows registry.
 - b Use the Windows registry editor to modify the [HKEY_CLASSES_ROOT\Wow6432Node\telnet\shell\open\command] key with the following value:

Name	Type	Data
(default)	REG_SZ	"C:\Program Files\PuTTY\putty.exe" %l

%l (with a lowercase L) is the argument passed to telnet, usually an IP address or the fully-qualified domain name of a node.



In a .reg file, escape each quotation mark (") and backslash (\) character with a backslash (\) character.

- 4 Restart the web browser, and then, in the browser address bar, enter the telnet command:

```
telnet://<node>
```

<node> is the IP address or fully-qualified domain name of a node that runs the telnet server.

If you are prompted with a security warning, allow the action.

In Firefox, select the **Remember my choice for telnet links** check box.

Third-Party SSH Client (Standard Windows and Windows on Windows)

This procedure applies to the following cases:

- 32-bit Internet Explorer on a 32-bit or 64-bit operating system
- 32-bit Firefox on a 32-bit or 64-bit operating system
- 64-bit Internet Explorer on a 64-bit operating system

To configure a third-party SSH client for use by a web browser, follow these steps:

- 1 Obtain and install a third-party SSH client.

This procedure gives examples for the PuTTY client installed to C:\Program Files\PuTTY\putty.exe. The PuTTY client is available from <http://www.putty.org>.

- 2 Because PuTTY cannot correctly parse the "ssh://<node>" input, this example includes a script that strips the "ssh://" from the input argument. The script C:\Program Files\PuTTY\ssh.js contains the following commands:

```
host = WScript.Arguments(0).replace(/ssh:/, "").replace(/\/\/g, "");
shell = WScript.CreateObject("WScript.Shell");
shell.Run("\"c:\\Program Files\\PuTTY\\putty.exe\" -ssh " + host);
```



This script was created for this example and is not included with PuTTY.

- 3 Define the ssh protocol.
 - a Back up the Windows registry.
 - b Use the Windows registry editor to add the [HKEY_CLASSES_ROOT\ssh] key with the following values:

Name	Type	Data
(default)	REG_SZ	URL:ssh Protocol
EditFlags	REG_DWORD	2
FriendlyTypeName	REG_SZ	Secure Shell
URL Protocol	REG_SZ	<i>no value</i>

- 4 Set file association for the URL:ssh Protocol file type.
 - a Back up the Windows registry.
 - b Use the Windows registry editor to modify the [HKEY_CLASSES_ROOT\ssh\shell\open\command] key with the following value:

Name	Type	Data
(default)	REG_SZ	"C:\Windows\System32\WScript.exe" "C:\Program Files\PuTTY\ssh.js" %1

%1 (with a lowercase L) is the complete ssh argument, including the protocol specification. The ssh.js script passes the ssh target to PuTTY.



In a .reg file, escape each quotation mark (") and backslash (\) character with a backslash (\) character.

- 5 Restart the web browser, and then, in the browser address bar, enter the ssh command:

ssh://<node>

<node> is the IP address or fully-qualified domain name of a node that runs the telnet server.

If you are prompted with a security warning, allow the action.

In Firefox, select the **Remember my choice for ssh links** check box.

- 6 Create a new NNMi console menu item that invokes the ssh protocol as described in [Configure a New Menu Item that uses the ssh Protocol](#) on page 272.

Configure Firefox to use Telnet or SSH on Linux

On the Linux operating system, define the telnet or ssh protocol, and then configure Firefox to use the new protocol.

To complete any of the procedures in this section, you must have administrative privileges on the computer.

For more information, see http://kb.mozillazine.org/Register_protocol.


Telnet on Linux

To configure Firefox on the Linux operating system to use the telnet protocol, follow these steps:

- 1 Define the telnet protocol.
 - a Create the `/usr/local/bin/nmmtelnet` file with the following contents:

```
#!/bin/bash
#
# Linux shell script called by Firefox in response to
# telnet:// URLs for the NNMi telnet menu.
#
address=`echo $1 | cut -d : -f 2 | sed 's;/;/g'`
port=`echo $1 | cut -d : -f 3`
exec /usr/bin/xterm -e telnet $address $port
```
 - b Set the script permissions to be executable by everyone:

```
chmod 755 /usr/local/bin/nmmtelnet
```
- 2 Configure Firefox preferences for telnet.
 - a In the Firefox address bar, enter: **about:config**
 - b In the preference list, right-click, click **New**, and then click **Boolean**.
 - c Enter the preference name: **network.protocol-handler.expose.telnet**
 - d Select the preference value: **false**
- 3 Configure Firefox to use the newly-defined protocol.
 - a Browse to a telnet link.

 You can create a simple HTML file containing the link, or you can use **Actions > Telnet... (from Client)** in the NNMi console. Typing the link directly into the address bar does not have the same effect.
 - b In the Launch Application window, click **Choose**, and then select `/usr/local/bin/nmmtelnet`.
 - c Select the **Remember my choice for telnet links** check box.

Secure Shell on Linux

To configure Firefox on the Linux operating system to use the ssh protocol, follow these steps:

- 1 Define the ssh protocol.
 - a Create the `/usr/local/bin/nmssh` file with the following contents:

```
#!/bin/bash
#
# Linux shell script called by Firefox in response to
# ssh:// URLs for the NNMi SSH menu.
#
address=`echo $1 | cut -d : -f 2 | sed 's/;/;g'`
port=`echo $1 | cut -d : -f 3`
exec /usr/bin/xterm -e ssh $address $port
```
 - b Set the script permissions to be executable by everyone:

```
chmod 755 /usr/local/bin/nmssh
```
- 2 Configure Firefox preferences for SSH.
 - a In the Firefox address bar, enter: **about:config**
 - b In the preference list, right-click, click **New**, and then click **Boolean**.
 - c Enter the preference name: **network.protocol-handler.expose.ssh**
 - d Select the preference value: **false**
- 3 Configure Firefox to use the newly-defined protocol.
 - a Browse to an SSH link.



You can create a simple HTML file containing the link, or you can use the new SSH menu item that you defined in the NNMi console. Typing the link directly into the address bar does not have the same effect.

- b In the Launch Application window, click **Choose**, and then select `/usr/local/bin/nmssh`.
- c Select the **Remember my choice for ssh links** check box.

Example Files for Changing the Windows Registry

If many NNMi users need to use the telnet or ssh protocols to access managed nodes from the NNMi console, you might be able to automate the Windows registry updates with one or more `.reg` files. This section contains example `.reg` files on which you can base the creation of your own `.reg` files. Note that the registry keys are located in a different path for running 32-bit applications on 64-bit versions of Windows than they are for when the application and operating system match.

For more information, see the Microsoft article at <http://support.microsoft.com/kb/310516>.

Example nntelnet.reg

This registry content example applies to [Windows Operating System-Provided Telnet Client](#) on page 274.

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet
Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL]
"iexplore.exe"=dword:00000000

[HKEY_CLASSES_ROOT\telnet\shell\open\command]
@="\"C:\Windows\system32\rundll32.exe\"
\"C:\Windows\system32\url.dll\",TelnetProtocolHandler %1"
```

Example nnmputtytelnet.reg

This registry content example applies to [Third-Party Telnet Client \(Standard Windows\)](#) on page 275.

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet
Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL]
"iexplore.exe"=dword:0c000000

[HKEY_CLASSES_ROOT\telnet\shell\open\command]
@="\"C:\Program Files\PuTTY\putty.exe\" %1"
```

Example nntelnet32on64.reg

This registry content example applies to [Third-Party Telnet Client \(Windows on Windows\)](#) on page 276.

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Internet
Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL]
"iexplore.exe"=dword:00000000

[HKEY_CLASSES_ROOT\Wow6432Node\telnet\shell\open\command]
@="\"C:\Program Files\PuTTY\putty.exe\" %1"
```

Example nnmssh.reg

This registry content example applies to [Third-Party SSH Client \(Standard Windows and Windows on Windows\)](#) on page 277.

```
Windows Registry Editor Version 5.00

[HKEY_CLASSES_ROOT\ssh]
@="URL:ssh Protocol"
"EditFlags"=dword:00000002
"FriendlyTypeName"="Secure Shell"
"URL Protocol"=""

[HKEY_CLASSES_ROOT\ssh\shell\open\command]
@="\"C:\Windows\System32\WScript.exe\" \"c:\Program Files\PuTTY\ssh.js\" %1"
```


Changing the NNMi Management Server

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 an 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 chapters in the [Integrations with NNMi](#) section.

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:

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

Ensure that the requirements listed in [Different System Restore](#) on page 260 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 284, and then begin data collection fresh on the target system.

Moving the NNMi Configuration

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



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

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



The `nmconfigexport.ovpl` command does not retain SNMPv3 credentials. For more information, see the `nmconfigexport.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 NNMi 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 `nmm.keystore` file stores the public key certificate that NNMi uses for encryption. The NNMi installation process creates the `nmm.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 `nmm.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 `nmm.keystore` File](#)
- [Task 3: Attempt to Locate the Original `nmm.keystore` File](#)
- [Task 4: If Available, Restore the Original `nmm.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 `nmm.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 `nmm.keystore` File](#)

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

Task 3: [Attempt to Locate the Original `nmm.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 the ovjboss process:

```
ovstop ovjboss
```

- 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 the ovjboss process:

```
ovstart ovjboss
```

- 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 an 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 prior to 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 an NNMi Management Server](#) on page 288.

- 4 Stop ovjboss:

```
ovstop ovjboss
```

- 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.

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.

For more information about the `server.xml` file and an example https connector block, see [Update the server.xml File](#) on page 86.

Task 5: Restart, Update, and Refresh Systems

- 1 Start ovjboss:

```
ovstart ovjboss
```

- 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 prior to changing the name of the NNMi management server overwrites the `nnm.keystore` file, thereby making the NNMi database inaccessible. If you need to 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 `ovjboss`:

```
ovstop ovjboss
```
- 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 `ovjboss`:

```
ovstart ovjboss
```

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.

Upgrading to NNMi 9.00

You can upgrade NNMi according to the information show in [Table 24](#). For best results, upgrade to NNMi 8.1x patch 7 or newer prior to upgrading to NNMi 9.00. The information shown in [Table 24](#) assumes you have NNMi 8.10 or newer installed on the NNMi management server.

Table 24 Supported NNMi Upgrades

NNMi Version	Upgrade to NNMi 9.0x
8.10	Supported
8.10 Patch 1	Supported
8.1x Patch 2 (NNMi 8.11)	Supported
8.1x Patch 3	Supported
8.1x Patch 4 (NNMi 8.12)	Supported
8.1x Patch 5 (NNMi 8.13)	Supported
8.1x Patch 6	Supported
8.1x Patch 7	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 [Unconfiguring NNMi from an HA Cluster](#) on page 225 or [Application Failover and Upgrading to NNMi 9.00](#) on page 192 in the *NNMi Deployment Reference*.

See [Table 25](#) to view the supported upgrade paths to NNMi 8.10. If you have a version of NNM older than NNMi 8.10 installed, upgrade directly to NNMi 9.0x Patch 1.

Table 25 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](#), which describes the following upgrade scenario:
 - Upgrading from NNMi 8.0x or 8.1x to NNMi 9.00 on the same hardware and operating system.
- [Upgrading to a Different NNMi Management Server](#), which describes the following upgrade scenario:
 - Upgrading from NNMi 8.0x or 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 prior to 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 9.00 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 9.0x Patch 1.

This chapter contains the following topics:

- Start from NNMi 8.0x
- Upgrade an Existing NNMi Management Server to NNMi 9.0x Patch 1

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 Patch 1](#).

Upgrade an Existing NNMi Management Server to NNMi 9.0x Patch 1

Read the NNMi 9.0x Patch 1 *Preinstallation Checklist* chapter in the *NNMi Installation Guide* and [Additional Upgrade Information](#) on page 307 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 Patch 1. 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 `nnmconfigexport.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 `nnmconfigexport.ovpl` or `nnmconfigimport.ovpl` reference pages, or the UNIX manpages.



Never edit a file exported with the `nnmconfigexport.ovpl` script before using the `nnmconfigimport.ovpl` script to import the file.

- 4 Install NNMi 9.0x Patch 1 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 version 9.00 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](#).


Upgrade to a Different NNMi Management Server

Read the NNMi 9.0x Patch 1 *Preinstallation Checklist* chapter in the *NNMi Installation Guide* and [Additional Upgrade Information](#) on page 307 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 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 9.0x Patch 1. For information, see [Migrating NNMi Oracle Data](#) on page 305.

- 1 As a precaution, back up the existing (source) NNMi 8.1x management server using the `nmbackup.ovpl` script. Label this backup for 8.1x. For more information, see the *nmbackup.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 *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 9.0x Patch 1 is working correctly on the source NNMi management server.
 - 5 Back up NNMi 9.0x Patch 1 on the source NNMi management server using the `nmbackup.ovpl` script. Label this backup for NNMi 9.0x Patch 1. 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 9.0x Patch 1.
 - 6 Install NNMi 9.00 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.
 - 9 Verify that the information from the target NNMi management server migrated successfully from the existing NNMi management server.

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 prior to 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

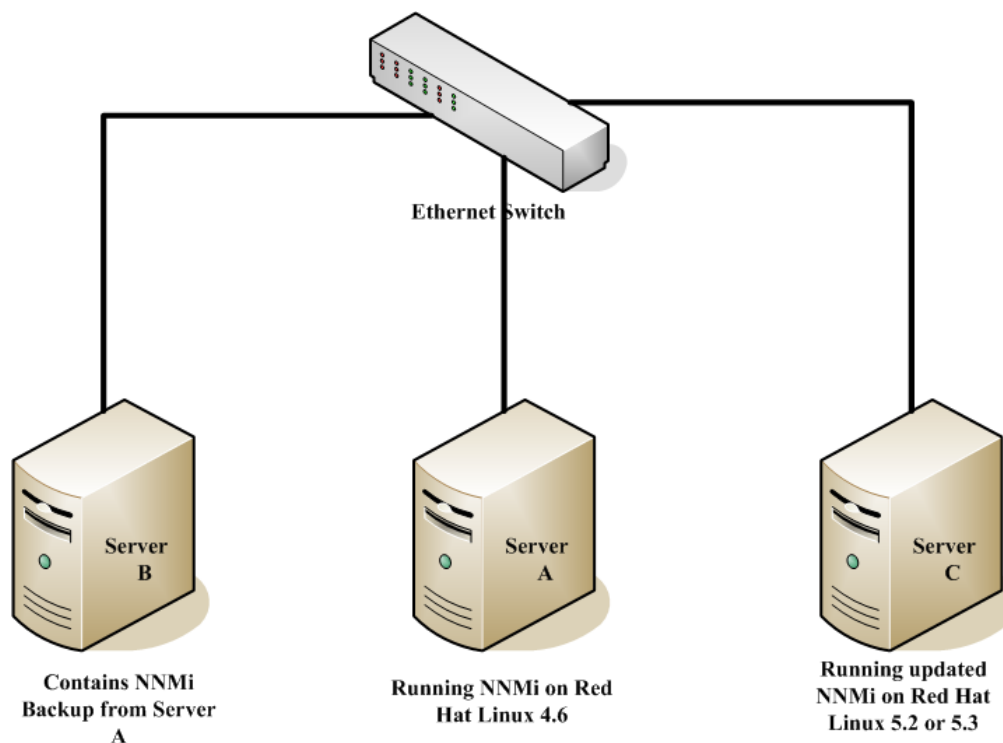
To complete the following steps, you must have NNMi 8.1x patch 6 or later running on a Linux Red Hat 4.6 server. To check the NNMi version number, note the current patch level in the **About Network Node Manager i-series** window. Verify that the version is 8.13.006 or later. If the version is earlier 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](#) on page 255 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](#) on page 255 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 an NNMi Management Server](#) on page 288.

Migrating NNMi Oracle Data

Suppose you need to 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 need to upgrade to NNMi 9.00.
- NNMi 9.00 connected to an Oracle 10G or Oracle 11G database.

The Oracle database instance migration you need to 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 9.00, perform that migration now, following the installation instructions in the *HP Network Node Manager i Software Installation Guide*.

If you are already using NNMi 9.00, 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 9.0x Patch 1. 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 9.0x Patch 1 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 9.0x Patch 1, 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 need 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 `nmsetdampenedinterval.ovpl` script to adjust the dampening interval. This script sets the dampening interval for all management event configurations. See the `nmsetdampenedinterval.ovpl` reference page or the UNIX `manpage` for more information.
 - After upgrading, the `nmsetdampenedinterval.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 9.0x Patch 1. After upgrading, run the `nmsetdampenedinterval.ovpl` script to apply this value across NNMi.

- If you upgrade to NNMi 9.0x Patch 1, 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 `nmsetdampenedinterval.ovpl` script.

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

- NNMi 9.0x Patch 1 does not include the NodeUp management event incident. The upgrade to NNMi 9.0x Patch 1 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 9.0x Patch 1 adds the **Calculate Status** setting to the **Node Group** configuration form. Upgrading to NNMi 9.0x Patch 1 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 9.0x Patch 1, 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 9.0x Patch 1 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.

Upgrading from NNM 6.x/7.x

This section contains the following chapters:

- Product Comparison
- Upgrading from NNM 6.x/7.x
- Integrating NNM 6.x or NNM 7.x with NNMi

Product Comparison

This chapter describes the key differences between HP Network Node Manager (NNM) 6.x/7.x and HP Network Node Manager i Software. If you are familiar with a prior version of NNM, refer to this chapter as you plan and configure NNMi. If you are new to NNMi, you do not need to read this chapter.

This chapter contains the following topics:

- Network Discovery
- Status Monitoring
- Customizing Event Monitoring

Network Discovery

Discovery controls the network elements (devices, nodes, and their components) that are added into the database. In NNMi, “inventory discovery” refers to the activity of finding new nodes and “Layer 2 discovery” refers to the connectivity modeling previously performed by Extended Topology discovery.

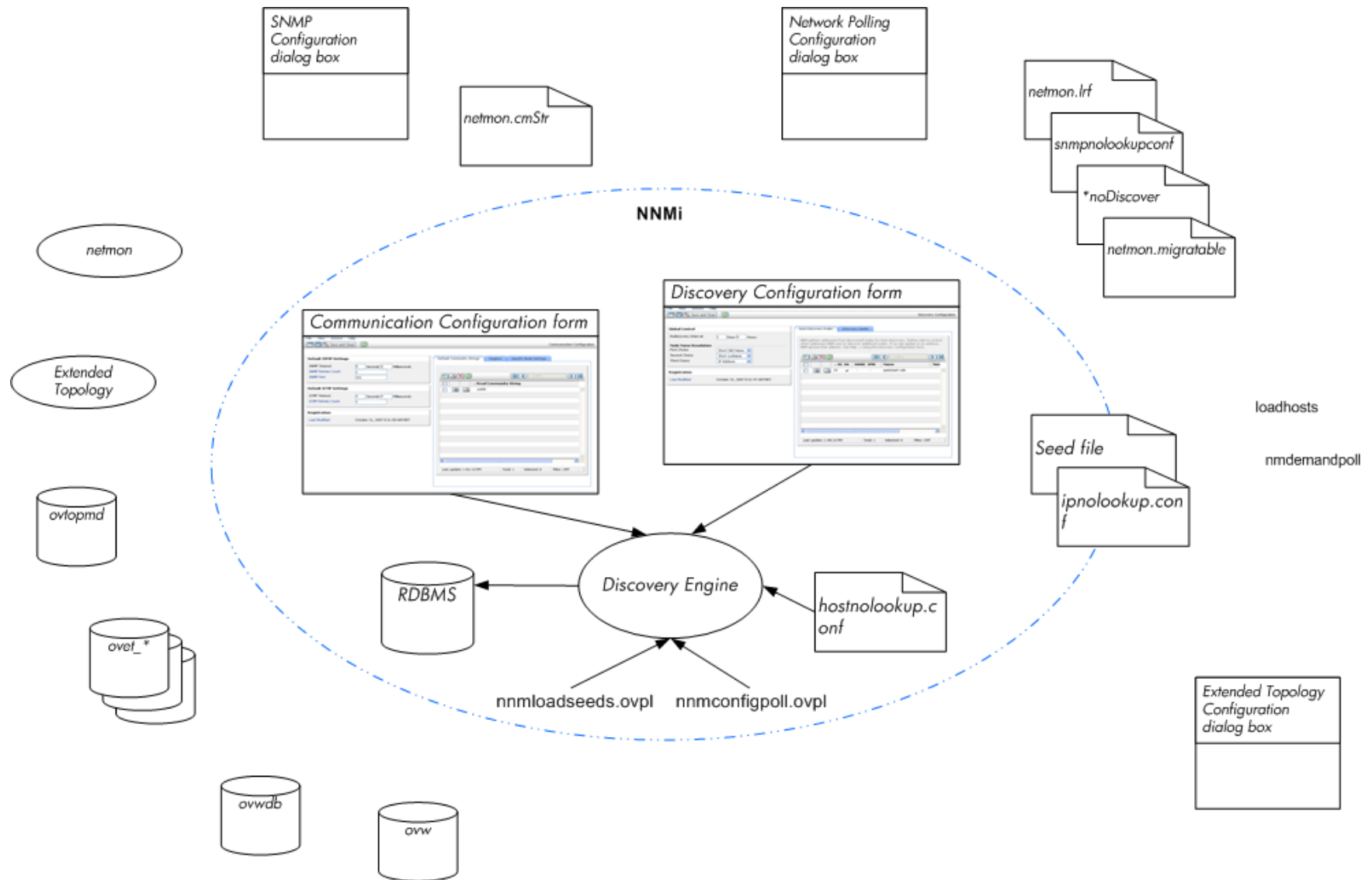
In NNM 6.x/7.x, by default, when NNM started, it used its own loopback address as a seed and started automatic discovery of the network to which it was directly connected (based on its own IP address and subnet mask). NNMi allows the administrator control from the beginning. For NNMi auto-discovery, you define discovery regions based on IP address ranges and specify at least one seed device (usually a router) before any discovery takes place.

The center of [Figure 17](#) shows the tools, files, and commands used to configure discovery in NNMi. The perimeter of the figure shows the similar items for NNM 6.x/7.x.



The Extended Topology information applies to NNM 6.x with the Extended Topology add-on or NNM 7.x Advanced Edition only.

Figure 17 Discovery Configuration Elements



Key Concepts for Discovery

This section briefly describes the main areas of change from NNM 6.x/7.x to NNMi. For more information about NNMi discovery, see *Discovering Your Network* in the NNMi help.

- NNMi stores all information in one relational database.
- NNMi uses one consolidated discovery engine that is easy to configure.
- The NNMi Spiral Discovery process provides ongoing updates to the topology information as changes occur in your network. Topology changes (both inventory and Layer 2) can be discovered more frequently than the scheduled rediscovery interval.
- In NNMi, all discovered nodes are counted against the license limit regardless of the management mode (MANAGED, UNMANAGED, or OUT OF SERVICE). You cannot discover nodes beyond the license limit.
- Auto-discovery has the same meaning in NNMi as in NNM 6.x/7.x, but the configuration approach is different.
 - In NNMi, you define the auto-discovery boundaries, provide at least one IP address seed, and then let discovery run.
 - NNMi auto-discovery uses an expanding model that is easy to control. NNMi auto-discovery finds and manages all routers, switches, and subnets within the boundaries that you provide. You specify the additional device types that NNMi should discover and manage.



- By default, non-SNMP nodes are *not* discovered in NNMi.
- Seeded discovery has the same meaning in NNMi as in NNM 6.x/7.x, but the configuration approach is different.
 - In NNMi, you can specify discovery seeds in the user interface.
 - You can use your NNM 6.x/7.x seed files in NNMi without modification.
 - The NNMi `nmmloadseeds.ovpl` command replaces the NNM 6.x/7.x `loadhosts` command.
- The NNMi configuration poll (`nmmconfigpoll.ovpl`) replaces the NNM 6.x/7.x demand poll (`nmdemandpoll`) for determining device configuration information.

Status Monitoring

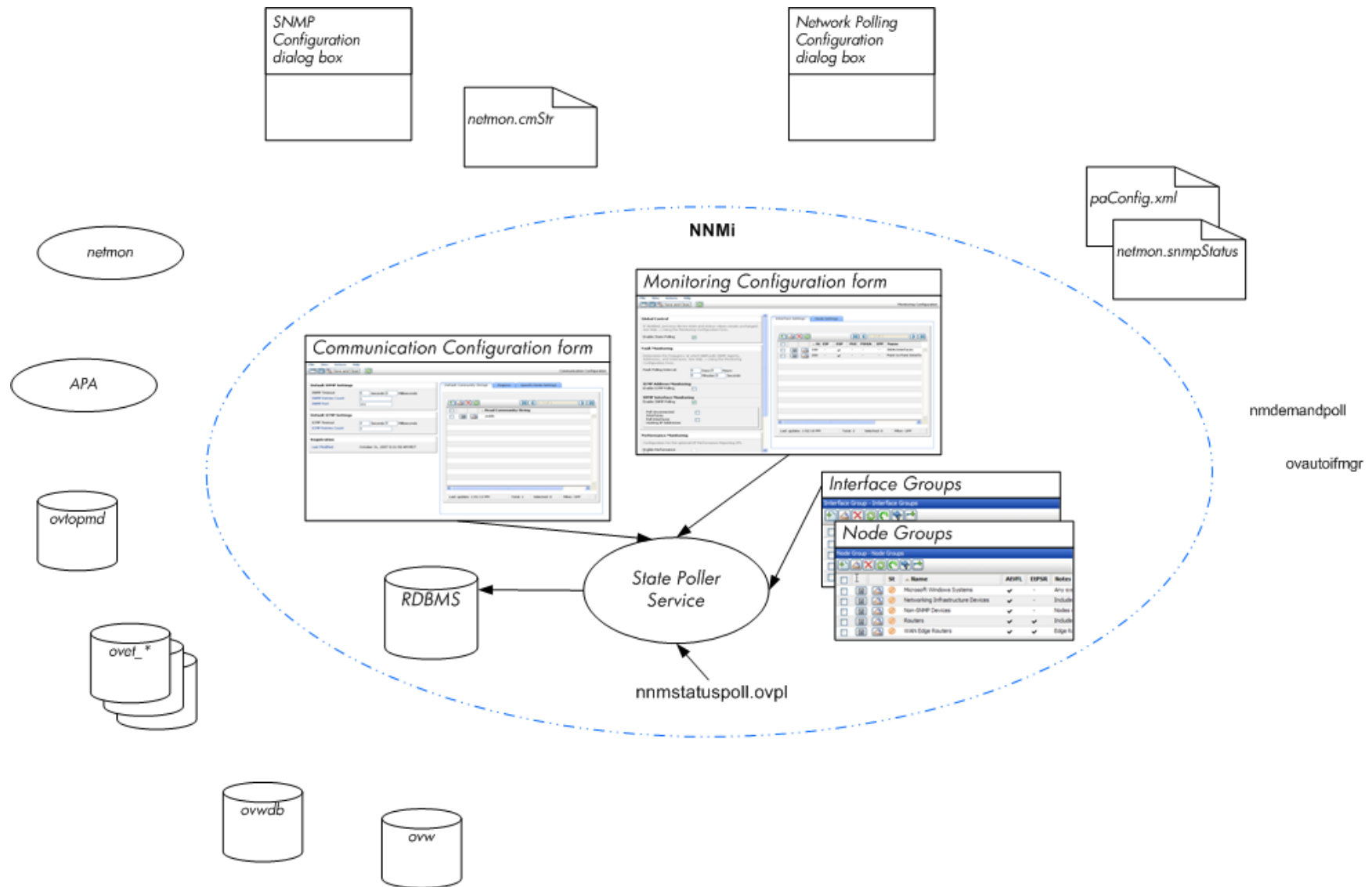
Status monitoring ensures that your network visualization is up-to-date in terms of devices or components that may have faults. When an element fails a poll, NNMi investigates the cause and issues a root cause alarm to the Incident Browser.

The center of [Figure 18](#) shows the tools, files, and commands used to configure status monitoring in NNMi. The perimeter of the figure shows the similar items for NNM 6.x/7.x.



The APA information applies to NNM 7.x Advanced Edition only.

Figure 18 Monitoring Configuration Elements



Key Concepts for Status Monitoring

This section briefly describes the main areas of change from NNM 6.x/7.x to NNMi. For more information about NNMi status monitoring, see *Monitoring Network Health* in the NNMi help.

- Configuration is now completed through the user interface.
- NNMi node groups and interface groups replace topology filters.
 - Groups can be filtered on a pre-defined set of attributes only.
 - Groups cannot be connected with Boolean operators.
 - Node groups use device filters instead of relying on sysObjectId wildcards.
 - Interface groups can be restricted based on the group of the containing node and the interface type.
- ICMP polling is disabled by default.
- Broad controls make it easier to exclude uninteresting interfaces.
- Monitoring settings are matched from most specific to most general: (1) interface settings, (2) node settings, (3) defaults.
- To change a monitoring behavior across the system, change all settings at all levels.
- The NNMi status poll (**Actions > Status Poll** or `nnmstatuspoll.ovpl`) replaces the NNM 6.x/7.x demand poll (`nmdemandpoll`) for determining device status.
- By default, NNMi only polls interfaces that are connected to another known interface through a Layer 2 connection. You can enable polling of unconnected interfaces and of interfaces that host IP addresses.

Customizing Event Monitoring

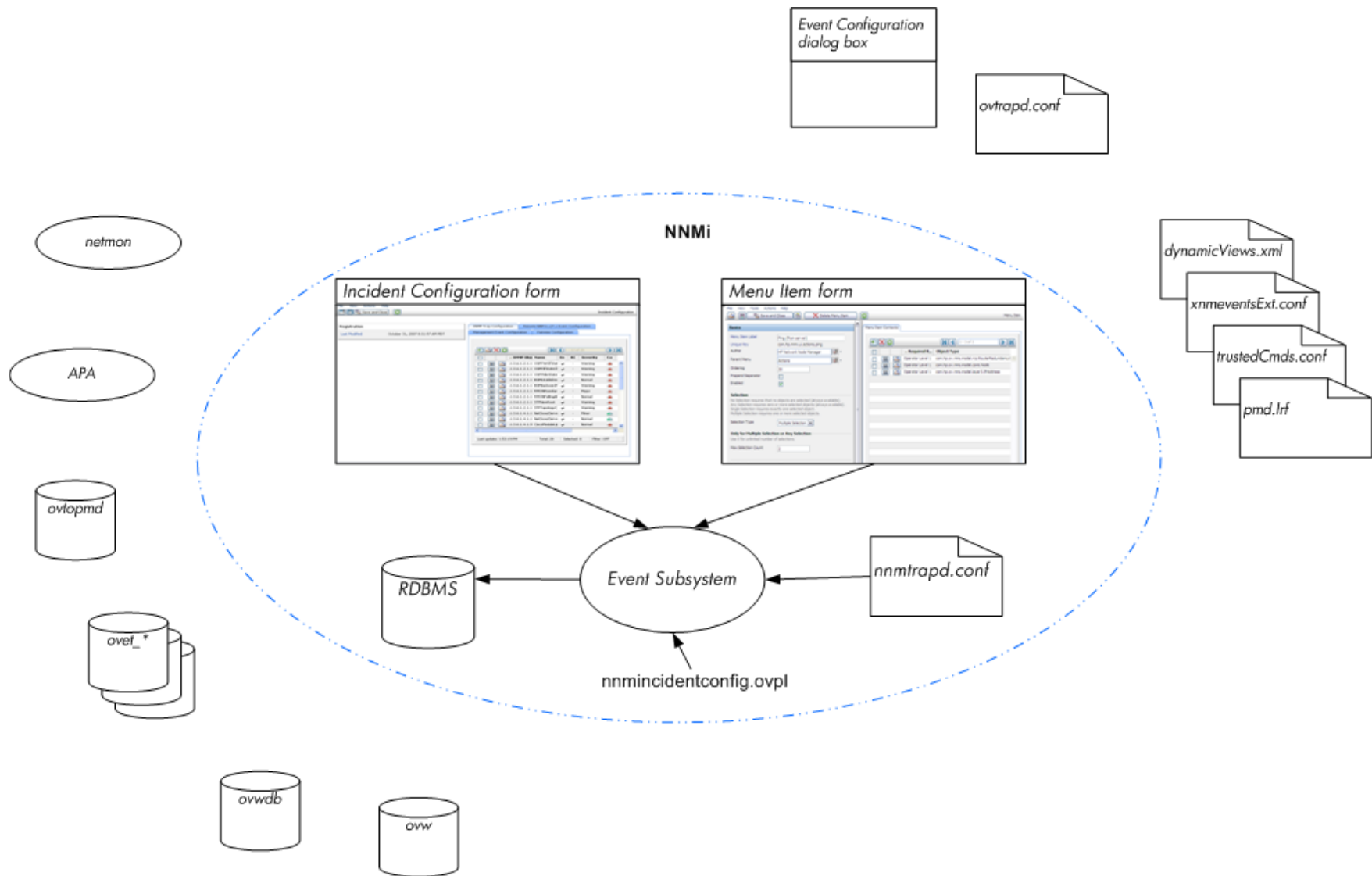
NNMi provides one centralized location, the incident views, where the management events, SNMP traps, and NNM 6.x/7.x forwarded events are visible to your team. You control which SNMP traps and NNM 6.x/7.x events are considered important enough to appear as incidents.

The center of [Figure 19](#) shows the tools, files, and commands used to configure event monitoring in NNMi. The perimeter of the figure shows the similar items for NNM 6.x/7.x.



The APA information applies to NNMi 7.x Advanced Edition only.

Figure 19 Event Monitoring Configuration Elements

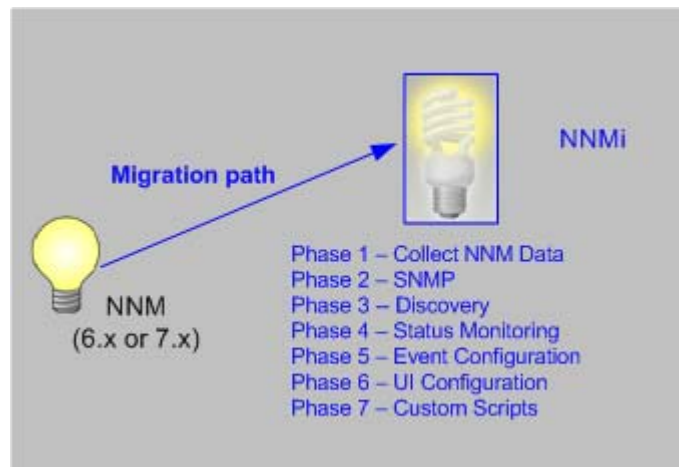


Key Concepts for Event Monitoring

This section briefly describes the main areas of change from NNM 6.x/7.x to NNMi. For more information about NNMi incidents, see *Configuring Incidents* in the NNMi help.

- In NNMi, the event subsystem is not used for inter-process communication and the volume of events is significantly reduced. The administrator no longer needs to configure whether each IPC message should be displayed or logged.
- NNMi receives only those traps for which it is configured. Unconfigured traps are filtered out of the event pipeline.
- NNMi displays all traps that it receives.
- Trap filters for the NNMi event subsystem processes are configured implicitly based on the selections in the **Incident Configuration** form.
- The NNMi `nmmincidentconfig.ovpl` command loads only the trap definitions from the named MIB files.
- NNMi provides pairwise, rate, and de-duplication correlations that occur in the event pipeline. (NNMi does not include the event correlation system (ECS).)
- In NNMi, you can configure actions that occur at any point in the lifecycle of an incident. Actions can be any script, executable, or Jython action.
- The NNMi URL Actions configuration replaces the `dynamicViews.xml` and `xnmeventsExt.conf` configuration files for defining actions that can be taken as a result of an event.

Upgrading from NNM 6.x/7.x



This chapter provides a basic path for upgrading from HP Network Node Manager 6.x or 7.x to the most recent version of the HP Network Node Manager i Software, as indicated in the footer of this document. This basic path should serve the needs of most users. This chapter does not cover advanced upgrade topics or customizations; consulting services are available to meet your needs in these areas.

This chapter uses 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).

This chapter makes the following assumptions:

- You have installed NNMI following the instructions in the *NNMI Installation Guide*.
- You have reviewed the concepts described in the NNMI help and the deployment information in this guide for a general understanding of NNMI functions.
- You understand how to use the NNMI console.

The information in this chapter will be updated frequently as tools that help with the upgrade process are released and as NNMI evolves.

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

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

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

A Fresh Beginning

Many NNM installations have been in place for several generations of software and in a variety of networking environments. Users who began with NNM 4.x or 5.x, in a routed world, might be carrying forward excess baggage that really does not apply to the current network structure. If your NNM installation is more than 2 years old, seriously consider using this opportunity to begin with a fresh 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 fresh installation of NNMi, install NNMi by following the instructions in the *NNMi Installation Guide*. Then consider the more complex deployment tasks presented in other chapters of this *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 365.

Table 26 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 26 Upgrade Continuum

Phase	Simplest Approach	Most Detailed and Thorough Approach
Collect Data from NNM	<ol style="list-style-type: none"> 1 Use the NNMi-provided tools on the NNM management station. 2 Copy the collected data to the NNMi management server. 	<ol style="list-style-type: none"> 1 At each upgrade phase, gather the appropriate NNM configuration data by hand. 2 Copy the collected data to the NNMi management server.
SNMP Information	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"> 1 Export all community strings currently in use. 2 Modify the data file and import the contents to NNMi as specific node community strings.
Discovery	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"> 1 Determine how NNM and <code>netmon</code> find nodes (seeds, loadhosts, filters, other tools). 2 Replicate this approach as closely as possible with seeds and auto-discovery rules.
Status Monitoring	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"> 1 Determine exactly what polling intervals and polling policies were used by NNM and <code>netmon</code> or APA for each group of nodes. 2 Implement NNMi node groups and interface groups to replicate the polling intervals and polling policies.
Event Configuration and Event Reduction	<ol style="list-style-type: none"> 1 Start with the default configuration from NNM. 2 Add the definitions for any custom traps from managed devices. 3 Add automatic actions as necessary. 	<ol style="list-style-type: none"> 1 Determine exactly what NNM customizations have been made for each trap and event type. 2 Customize each matching trap and event type on the NNMi system.
Graphical Visualization	<ol style="list-style-type: none"> 1 Import the NNM <code>ovw</code> containers. 2 Assign node groups to containers. <p>OR</p> <ol style="list-style-type: none"> 1 Import the NNM 7.x Advanced Edition container views. 2 Assign node groups to containers. 	<ol style="list-style-type: none"> 1 In the most inclusive NNM map, determine what is on each submap. 2 Create a node group for the contents of each NNM submap. 3 For each node group, create an NNMi map, add a background image, and place each node.
Custom Scripts	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 360.

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 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:

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

```
nnmmigration.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 356.



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:

archiveMigration.ovpl

This tool creates the `<hostname>.tar` file of 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` may 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\`
 - *UNIX:* `$NnmDataDir/tmp/`
- 2 In the `tmp` directory, create the `migration` and `<hostname>` directories in the following structure:
 - *Windows:* `%NnmDataDir%\tmp\migration\<hostname>\`
 - *UNIX:* `$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
```

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:

```
xnmssnmpconf -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\\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 328 (for the upgrade tool approach) or [step 1](#) on page 327 (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 NNMi's initial 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 Community Strings** tab.
- c Configure timeout, retries, and port.

The screenshot shows the NNMi Communication Configuration console. The left pane displays 'Default SNMP Settings' with options for 'Enable SNMP Address Rediscovery' and 'Enable SNMP GetBulk' checked. The 'SNMP Timeout' is set to 5 seconds and 0 milliseconds. Below this is an example of timeout and retries logic. Other settings include 'SNMP Retries Count' (2), 'SNMP Port' (161), and 'SNMP Minimum Security Level' (Community Only). The 'Default ICMP Settings' section shows 'ICMP Timeout' (2 seconds, 0 milliseconds) and 'ICMP Retries Count' (2). The right pane shows the 'Read Community Strings' tab with a list of community strings: 'ntpublic' and 'public'. The status bar at the bottom indicates 'Updated: 3/23/10 1:45:24 PM | Total: 2 | Selected: 0 | Filter: OFF | Auto refresh: OFF'.

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> mysnmp.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 mysnmp.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
- The preferred management address for specific nodes



When NNM selects a management address, it selects the lowest loopback address. NNMi 8.1x also selects the lowest loopback address. NNMi 8.0x selects the highest loopback address.

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 332.



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.

- 1 If available, edit the `ipnolookup.conf` and `hostnolookup.conf` files created by the `nnmmigration.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 330) 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.

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 need to identify these profiles at a later time.

- 2 In the NNMi console, select **Device Profiles** from the **Configuration** workspace. Locate the entry by `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

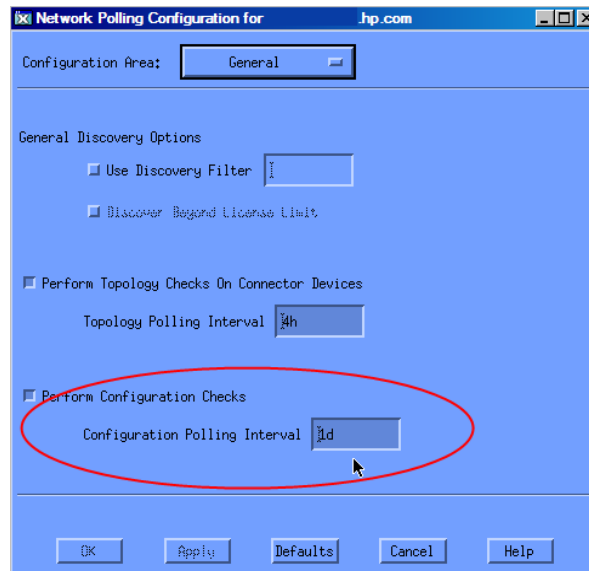
The NNM discovery processes can run independently. To upgrade discovery to NNMi, you only transfer the **interval** at which NNM discovers nodes.

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.

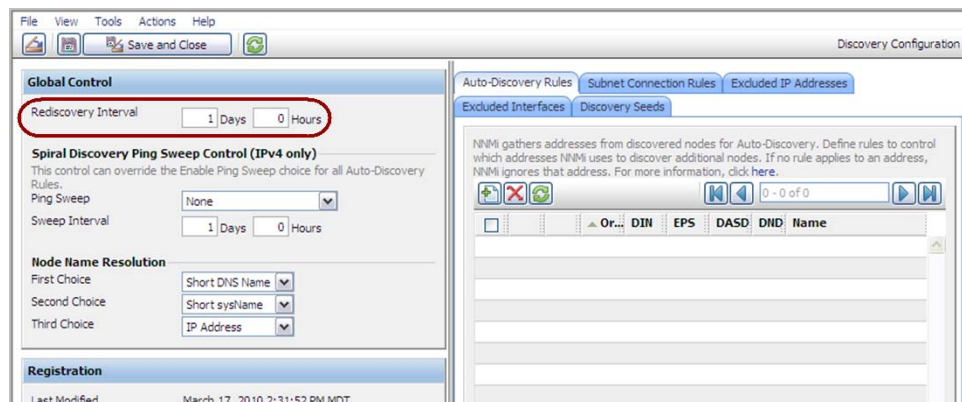
Gather from NNM

- 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

- 2 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 341
- Automatic discovery based on seeds and auto-discovery rules. Complete both of the following tasks:
 - [Configure Auto-Discovery Rules](#) on page 336
 - [Add Seeds to NNMi for Seeded Discovery](#) on page 341

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](#) on page 74.

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 and/or limit discovery 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)
- `netmon.interfaceNoDiscover`
- 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's 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 prior to 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:** %OV_DB%\nnmet\hosts.nnm
- **UNIX:** \$OV_DB/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 341.
 - 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 }
```

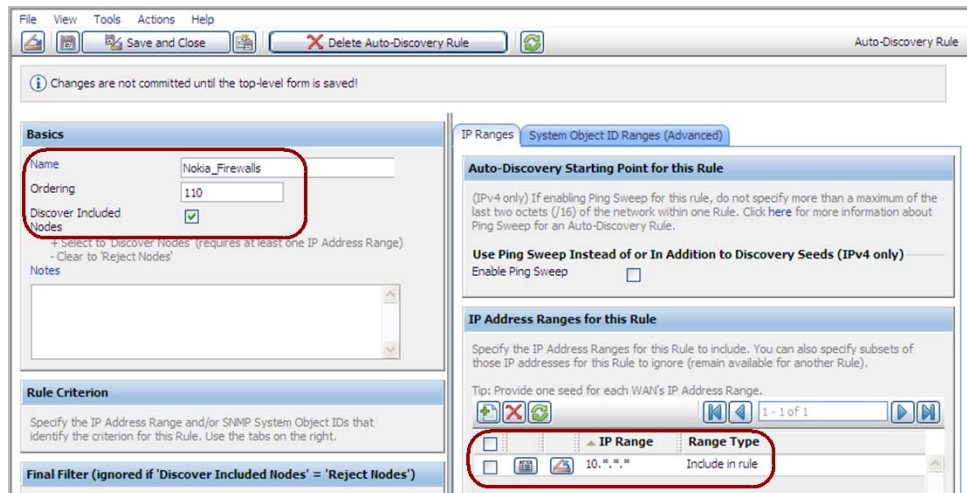
Replicate in NNMi

- 2 Enter the discovery filters in the NNMi console.

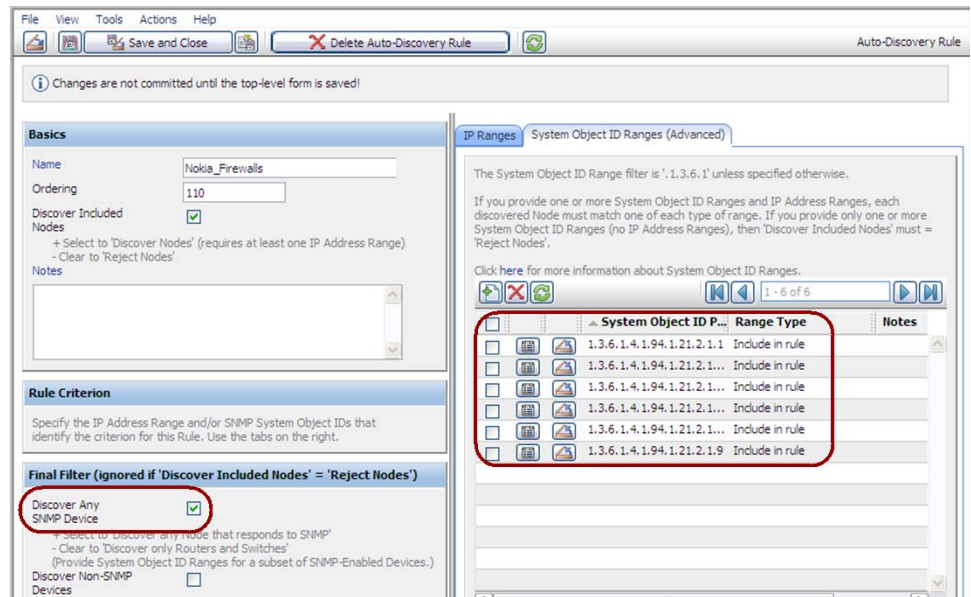
NNMi discovery filter entry example

For example, to transfer the NNM filter shown in the [NNM discovery filter example](#) on page 338 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.*.*.*.

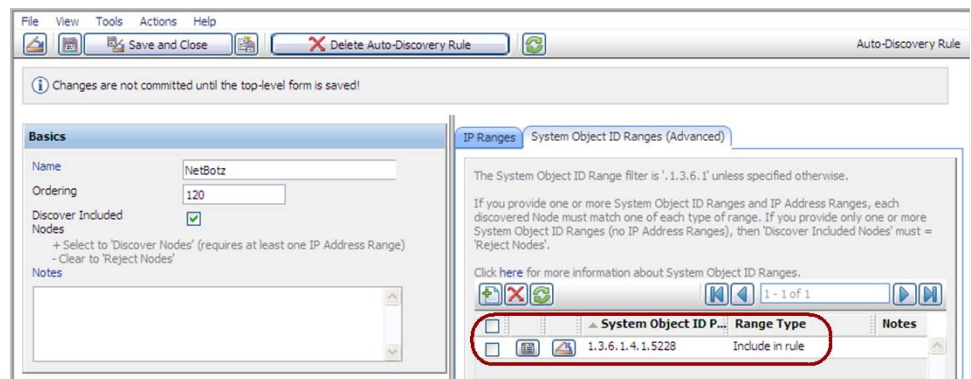
- a For Nokia firewalls, enter a rule name (Nokia_Firewalls), and then enter the network IP range 10.*.*.*.



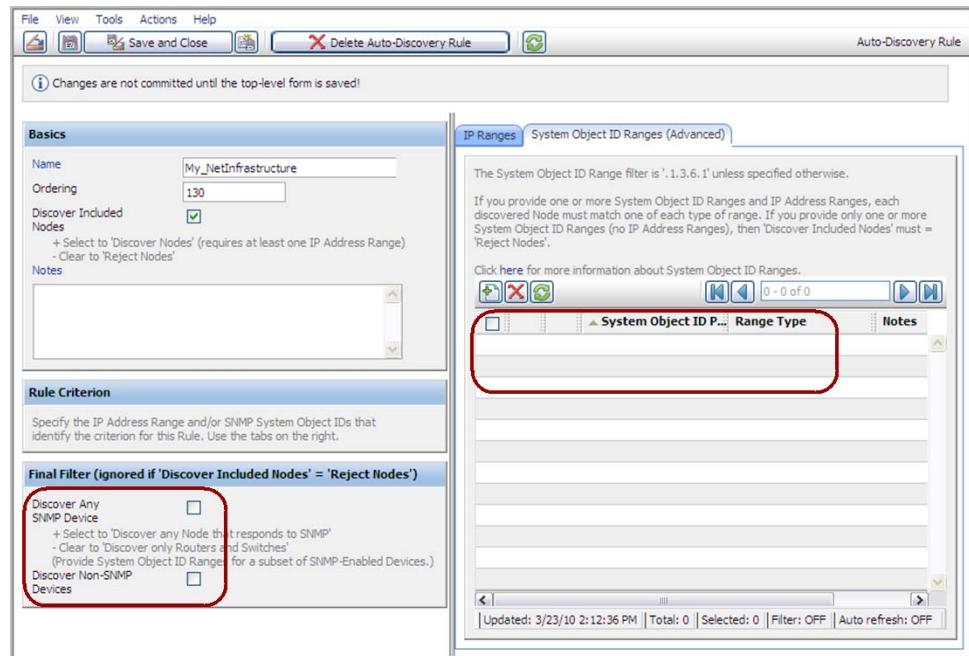
- b Enter each sysObjectID (do not enter the leading period), and then select the **Discover Any SNMP Device** check box. (By default, NNMi only discovers switches and routers. Because these devices might not be marked as switches or routers, select the **Discover Any SNMP Device** check box when specifying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

— *UNIX:*

```
$NnmDataDir/tmp/migration//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 342:

```
<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, or address level.

The following feature actions are no longer used in NNMi and are not transferable:

- Special handling for DHCP nodes
- Automatic deletion of a node that does not respond to polling
- Boards and aggregated ports, which are currently not modeled in NNMi and cannot have status monitoring

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>

```

```

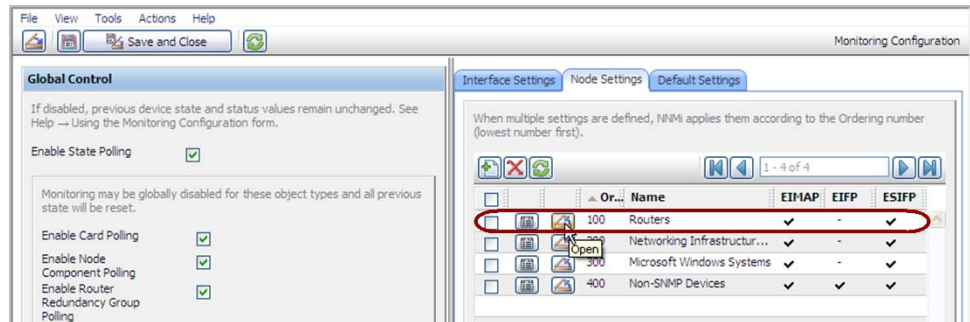
        The interval for which the device will be polled
        in seconds.
    </description>
    <varValue>
        <varType>Integer</varType>
        <value>300</value>
    </varValue>
</parameter>
.
.
.
</parameterList>
</classSpecification>

```

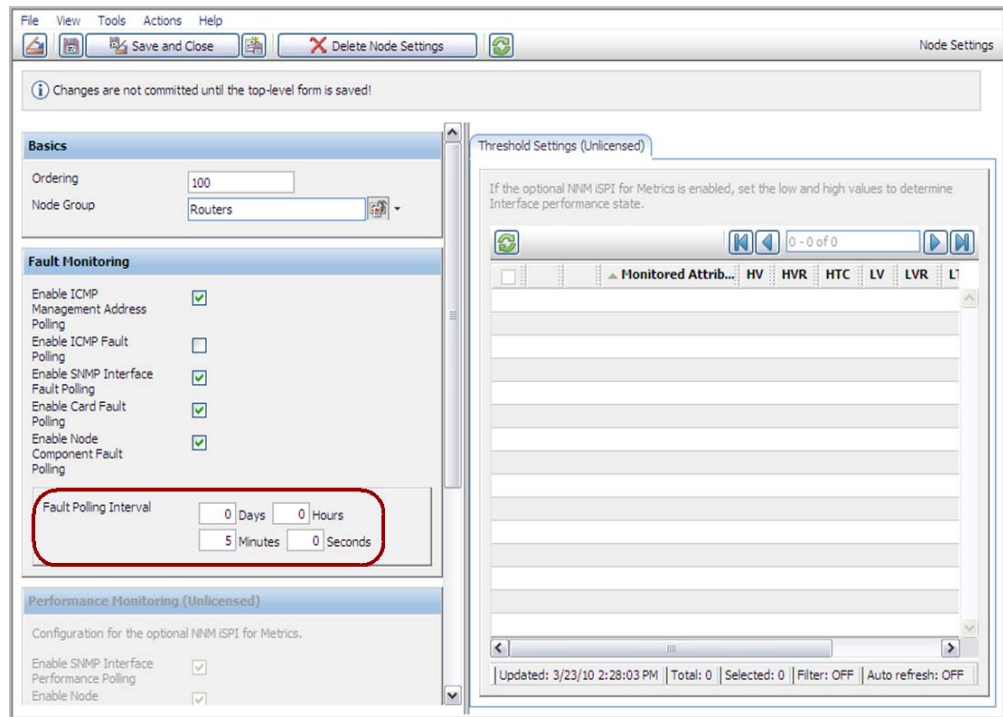
Replicate to NNMi **NNMi polling process**

NNMi status monitoring configuration is based on groups of nodes and/or groups of interfaces.

- 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

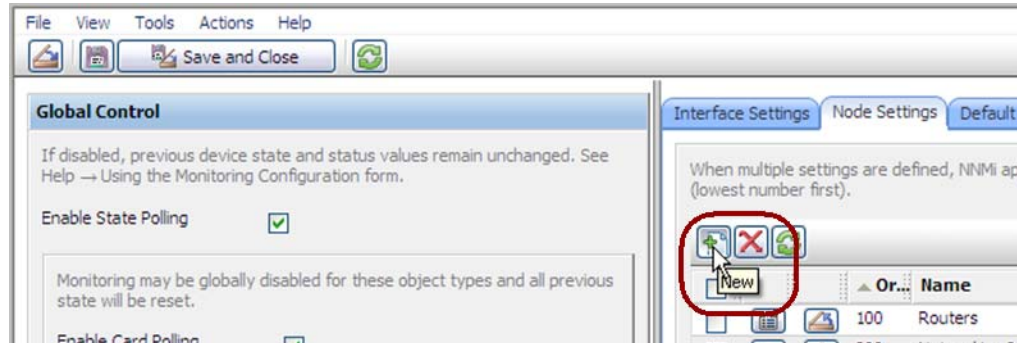
For example, to configure polling (using SNMP and ping) for a collection of VOIP routers, follow these steps:

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

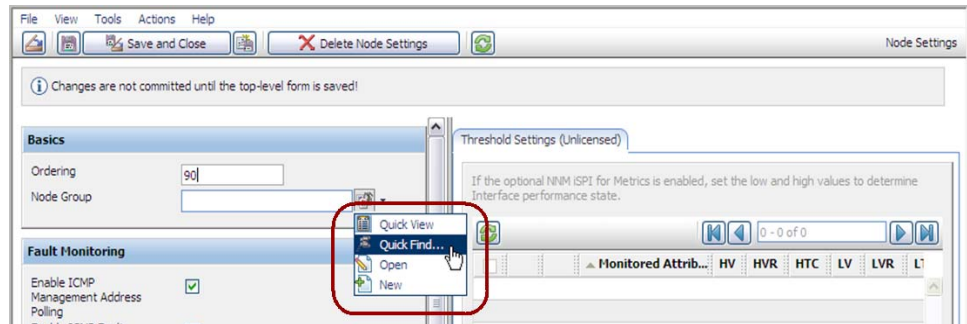
The screenshot shows the 'Node Group' configuration window in NNMi. The window title is 'Node Group'. The menu bar includes 'File', 'View', 'Tools', 'Actions', and 'Help'. The toolbar has icons for 'Save and Close' and 'Delete Node Group'. The main area is divided into several sections:

- Basics:** Name: VOIPRouters; Calculate Status: ; Status: No Status; Add to View Filter List: .
- Notes:** A text area for notes.
- Filter Editor:** A table with columns 'Attribute', 'Operator', and 'Value'. The first row contains 'hostname', '=', and '*voip*'. Buttons for 'Append', 'Insert', and 'Replace' are to the right.
- Filter String:** Shows the resulting filter string: 'hostname = *voip*'. Buttons for 'Insert', 'AND', 'OR', 'NOT', 'EXISTS', 'NOT EXISTS', and 'Delete' are to the right.
- Additional Information:** Text explaining filter operators and a link for more information.
- Additional Filters:** A section for adding more filters.
- Additional Nodes:** A section for adding additional nodes.
- Child Node Groups:** A section for adding child node groups.
- Status:** A section for setting the status.
- NNMi iSPI Performance:** A section for configuring performance metrics.

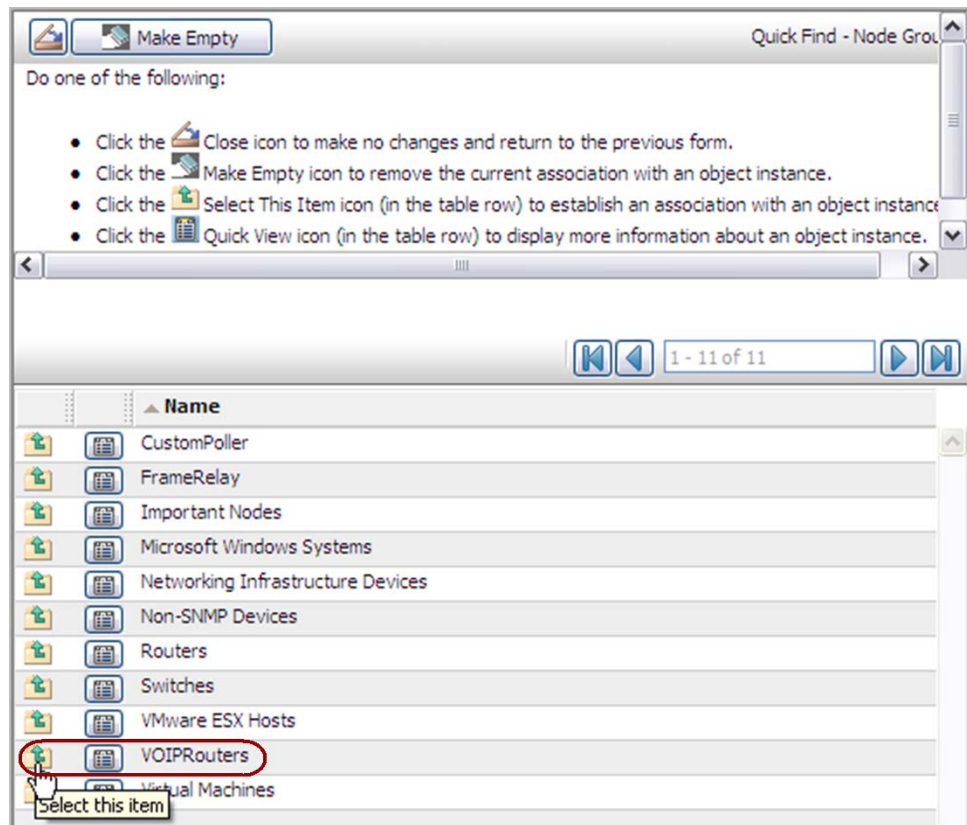
- 2 On the **Monitoring Configuration** form, add new node settings, as shown here.



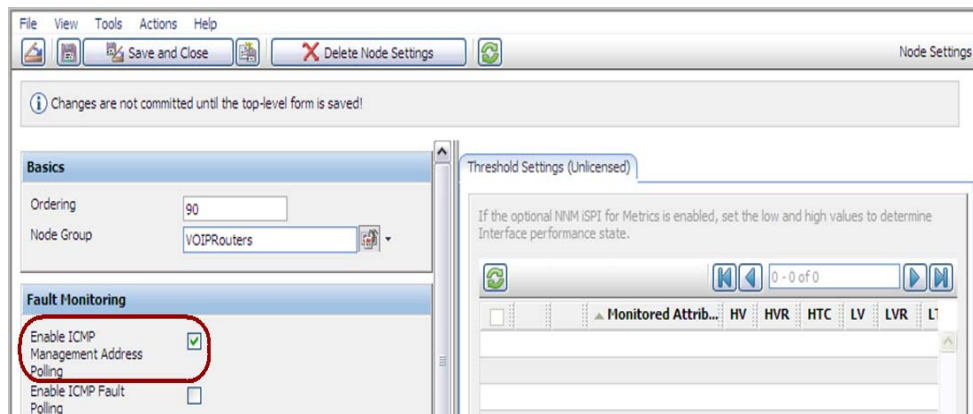
- 3 Specify an ordering value, and then select quick find for the **Node Group** field, as shown here.



- 4 Select the VOIPRouters node group, as shown here.



- Verify that the **Enable ICMP Management Address 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:

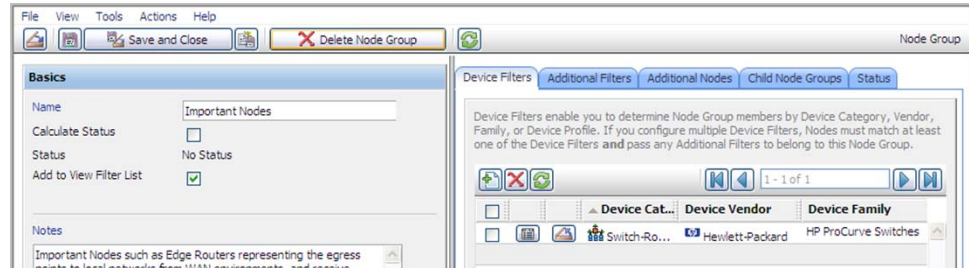
```
<HostIDs xmlns="http://www.hp.com/openview/NetworkTopology/
TopologyFilter" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xsi:schemaLocation="http://www.hp.com/openview/
NetworkTopology/TopologyFilter HostIDFile.xsd">
  <DNSName>router1.example.net</DNSName>
  <DNSName>router7.example.net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

Replicate to NNMi

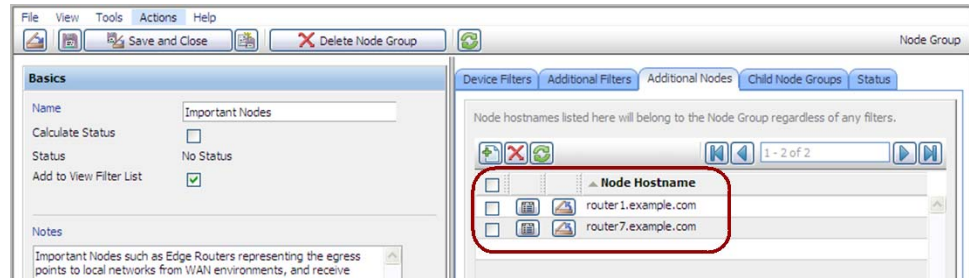
NNMi polling process

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

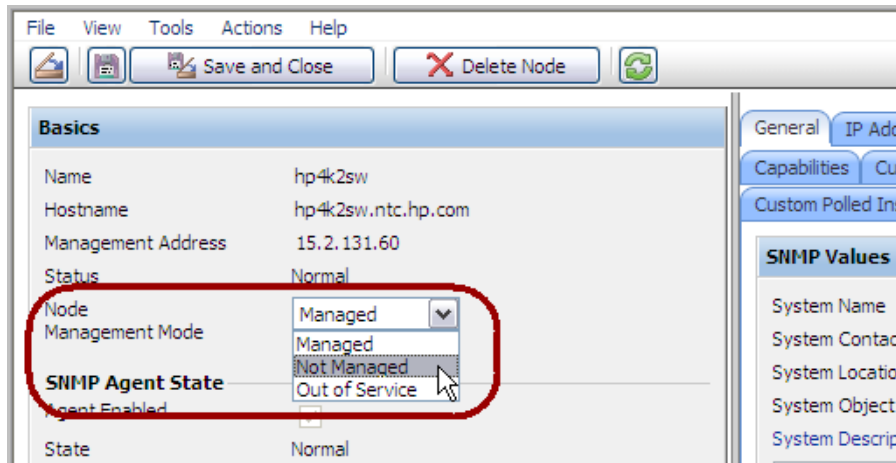


Exclude Objects from Status Polling

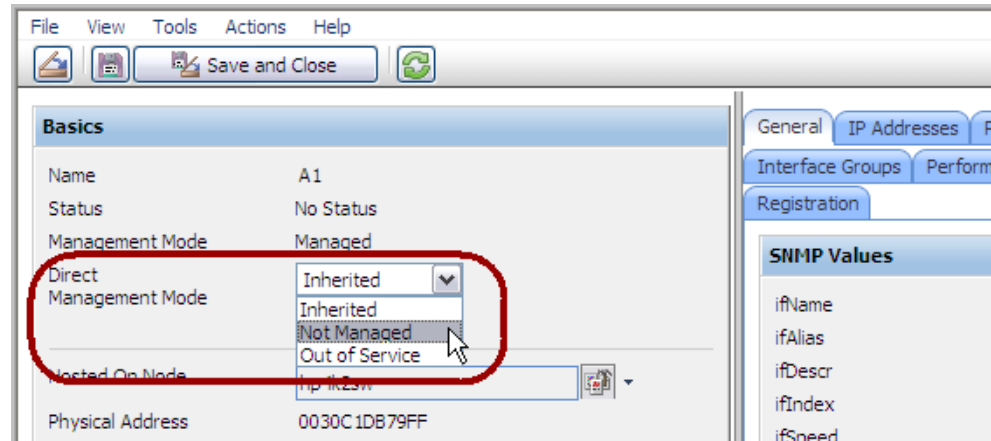
In NNM, most activities that stop nodes or interfaces from being monitored (set them to an UNMANAGED state) are completed through manual intervention in the NNM user interface.

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 various sources of events differently. Traps from devices and events forwarded from NNM management stations are in the SNMPv2c format. Incidents created within NNMi contain names only; they do not have event object identifiers. In addition, NNMi internal process communications use a new (non-trap) mechanism to significantly improve overall performance. NNMi does not have no format in trapd.conf messages for unrecognized events. Unrecognized events are now discarded. 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/<hostname>/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 need to identify these event definitions at a later time.

Manual approach

- 1 Download the vendor MIB files to the NNMi management server.
- 2 Run the following command for each MIB:

```
nmmincidentcfg.ovpl -loadTraps mibFile
```

 - If one MIB has a dependency on another MIB file, use the following command to preload the dependencies:

```
nmmincidentcfg.ovpl -loadMib mibFile
```

Alternatively, you can use the `nmmloadmib.ovpl` command, and then rerun `nmmincidentcfg.ovpl` with the `loadTraps` option.
 - To see which MIBs are already loaded, use the command:

```
nmmloadmib.ovpl -list
```

For more information, see the *nnmincidentcfg.ovpl* and *nnmloadmib.ovpl* reference pages, or the UNIX manpages.



These steps only load TRAP-TYPE and NOTIFICATION-TYPE MIB entries. NNMi does not use other MIB variables.

Enhance in NNMi

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

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, select **Incident Configuration** from the **Configuration** workspace. Then select the **Management Events** tab.
- 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, select **Incident Configuration** from the **Configuration** workspace. 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 9.0x Patch 1 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, select **Incident Configuration** from the **Configuration** workspace.
- 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 359.

- 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 > Documentation Library > Integrate NNMi Elsewhere with URLs** in the NNMi console.
- 3 In the NNMi console, select **User Interface Configuration** from the **Configuration** workspace.
- 4 On the **Menu Items** tab, 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.

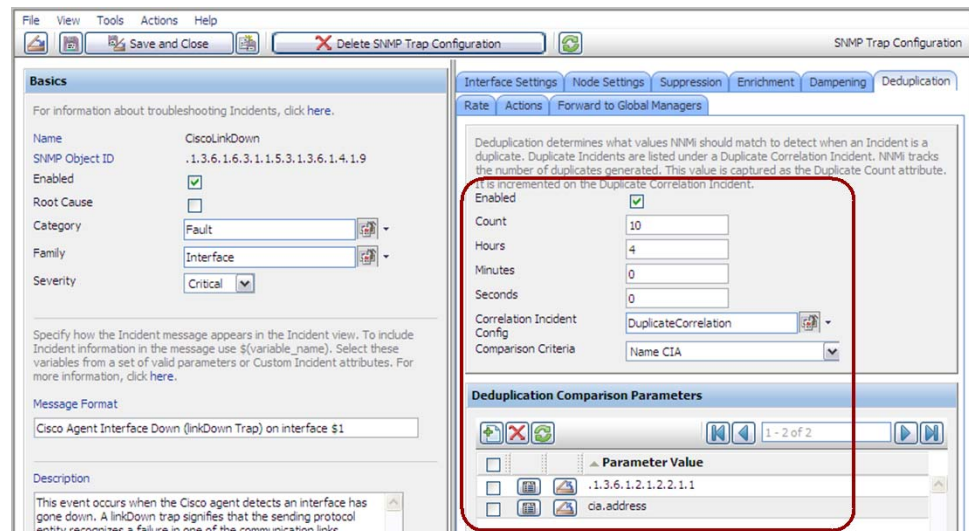
Gather from NNM

- 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, select **Incident Configuration** from the **Configuration** workspace.
- 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 **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, select **Incident Configuration** from the **Configuration** workspace.
- 3 Select the **Management Events** tab.
- 4 Open the incident type to be counted.
- 5 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 **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

Replicate to NNMi

- 1 Determine whether the PairWise correlation is in use in NNM.
- 2 Determine whether the Transient correlator is in use in NNM.
- 3 In the NNMi console, select **Incident Configuration** from the **Configuration** workspace.
- 4 On the **Pairwise Configuration** tab, 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

Replicate to NNMi

- 1 Determine whether the `ScheduledMaintenance` correlation is in use in NNM.
- 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 may 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 325.
- 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 `nmmnodegrouplist.csv` and `backgrounds.tar` files, which are available in the following location:

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

Replicate in NNMi

Upgrade tool approach

1 If you have not already done so, copy the `nmmnodegrouplist.csv` and `backgrounds.tar` files from the NNM management server to the following location:

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

2 Change to the following directory:

- *Windows:* `%NnmDataDir%\tmp\migration\<hostname>\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 > 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 may 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.

- 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 `nnm_topodump.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 `nnm_topodump.ovpl` command can generate textual output in a format very similar to that of the NNM `ovtopodump` command. Additionally, you might be able to replace other NNM commands in custom scripts with the `nnm_topodump.ovpl` command.

- 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:
 - `ovtopodump`
 - `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 27 lists the tools that are included in the data collection tools archive files.

Table 27 Upgrade Data Collection Tools

Tool	Description
<code>createMigrationDirs.ovpl</code>	Creates the directory structure to hold the upgrade data that will be collected from the NNM management station. For more information, see NNM Configuration Data Files on page 362.
<code>nmmigration.ovpl</code>	Collects the NNM configuration data. This tool is a wrapper script that runs most of the other tools described in this table.
<code>archiveMigration.ovpl</code>	Packs the collected data into a tar archive file (<code><hostname>.tar</code>) for easy transfer to the NNMi management server.

Table 27 Upgrade Data Collection Tools (cont'd)

Tool	Description
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>xnmsnmpconf -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 Table 29 .
trapdConfNodes.ovpl	Parses the <code>trapd.conf</code> file to create node lists (<code>EVENTS\NODES*</code>) that future upgrade tools might use.
nmmapmigration.ovpl	Parses the export file for an OVW map to identify node groups of the locations in that map (<code>nmnodegrouplist.csv</code> in the MAPS directory) and to collect the background image files that are used on location submaps (<code>backgrounds.tar</code> in the MAPS directory). Run this command separately from the <code>nmmigration.ovpl</code> 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 28](#) lists the contents of the `<hostname>` directory.

Table 28 File Structure of the Collected NNM Configuration Data

Directory	Contents
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 29 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 29 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.
nnmmibmigration.ovpl	Runs nnmincidentcfg.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 Table 27.

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

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

- You can forward events from NNM 6.x/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/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/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/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 366
- [Configure Remote View Launching](#) on page 370
- [Test the Integration](#) on page 373
- [Troubleshoot Event Forwarding](#) on page 376

Configure Event Forwarding

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

- [Step 1: Configure NNM 6.x/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/7.x Management Station to the NNMi Topology](#)
- [Step 4: \(Optional\) Save the Management Station Configuration](#)
- [Step 5: Verify NNM 6.x/7.x Incident Configuration in the NNMi Console](#)

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

On the NNM 6.x/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/7.x events to forward, see the events listed on the **Remote NNM 6.x/7.x Events** tab of the **Incident Configuration** form 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 368.

To configure an NNM 6.x/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 367.

- **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/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%\nnm8EventForwardDestinations.txt
- **UNIX:** \$OV_CONF/nnm8EventForwardDestinations.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/7.x Management Station to the NNMi Topology

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

If the NNM 6.x/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/7.x Incident Configuration in the NNMi Console

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

To view the NNMi default incident configurations, follow these steps:

- 1 In the NNMi console, select **Incident Configuration** from the **Configuration** workspace.
- 2 Click the **Remote NNM 6.x/7.x Events** tab.

This tab 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/7.x management station is not listed on the **Remote NNM 6.x/7.x Events** tab, 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/7.x are different from those in NNMi. For information about the relationship between the NNM 6.x/7.x alarm categories and the NNMi incident categories, see [Mapping Categories](#).

Mapping Categories

In NNM 6.x/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 30 lists the mapping of NNM 6.x/7.x alarm categories to NNMi incident categories that HP suggests:

Table 30 Suggested Category Mappings

NNM 6.x/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/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/7.x Management Station Entity in NNMi](#)
- [Step 3: \(Optional\) Configure Additional NNM 6.x/7.x Views](#)

Step 1: Install Java Plug-in

Although NNMi does not have any requirements for a Java Plug-in, NNM 6.x/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/7.x Management Station Entity in NNMi

Configure the NNMi management server to associate events received from the NNM 6.x/7.x management station to an entity in NNMi. This configuration enables the launching of NNM 6.x/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/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/7.x management station.


To set up an NNM 6.x/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/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/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/7.x management station.
 - Determine the custom incident attribute (CIA) of an incident that has been forwarded from the NNM 6.x/7.x management station.

- ▶ This method works only if you have already completed the procedures that are described in [Configure Event Forwarding](#) on page 366 and if a configured event has been generated on the NNM 6.x/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/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/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/7.x views.

Step 3: (Optional) Configure Additional NNM 6.x/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/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/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/7.x Dynamic Views from NNMi](#)

Test 1: Verify Event Forwarding

Under normal network conditions, NNM 6.x/7.x generally receives network events. The NNM 6.x/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/7.x management station to the NNMi management server, follow these steps:

- 1 On the NNM 6.x/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/7.x management station. For information about how to run this command, see [sendMsg.ovpl](#) on page 375.

Another approach is to generate or simulate a network fault on the NNM 6.x/7.x system. See [Generate Test Interface Down and Interface Up Events](#) on page 374.

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

The event that you generated from the NNM 6.x/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/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/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 may 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/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/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/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/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/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/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.



By default, the `OV_Message` incident is not configured in NNMi. To run this test, the `OV_Message` event in NNM 6.x/7.x must be configured to forward to the NNMi management server, and the `OV_Message` incident must be configured in the NNMi **Incident Configuration** form.

Test with Traps to NNM 6.x/7.x System

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

You can manually generate traps on the NNM 6.x/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/7.x system. For more information, see the *snmptrap* manpage.

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

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

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

- NNM 6.x/7.x Home Base
- NNM 6.x/7.x ovw
- NNM 6.x/7.x MIB Browser
- NNM 6.x/7.x Launcher
- NNM 6.x/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/7.x events in the **NNM 6.x/7.x Events** incident view, follow these steps to troubleshoot the problem:

- 1 On the NNM 6.x/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/7.x management station in the last *n* minutes.

- 2 If an expected event is not included in the `ovdumpevents` output, the event was not generated. See the NNM 6.x/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/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 *n* minutes.

- 5 For each expected event that is not included in the `nmmdumpevents` output, verify the configuration of that event in the **Event Configurator** window on the NNM 6.x/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/7.x to Forward Events to the NNMi Management Server](#) on page 366.

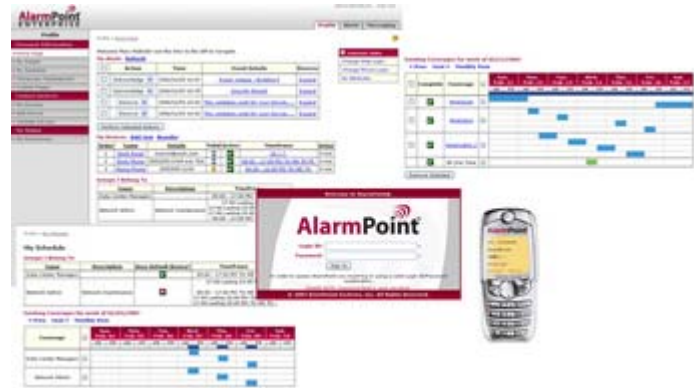
- 6 Repeat [step 5](#) until all expected events are included in the `nmmdumpevents` output on the NNMi management server.
- 7 In the NNMi console, examine the **NNM 6.x/7.x Events** incident view. If the results are not as expected, verify the incident configuration from the **Remote NNM 6.x/7.x Events** tab of the **Incident Configuration** form.

Integrations with NNMi

This section contains the following chapters:

- AlarmPoint
- CiscoWorks LAN Management Solution
- Clarus Systems ClarusIPC Plus⁺
- HP Asset Manager
- HP Business Availability Center My BSM
- HP Business Service Management Topology
- HP Universal CMDB
- HP Network Automation
- HP ProCurve Manager Plus
- HP RAMS MPLS WAN
- HP Systems Insight Manager
- nGenius Performance Manager
- NNMi Northbound Interface
- HP BSM Operations Management
- HP Operations Manager
- HP NNMi Integration Module for Netcool Software

AlarmPoint



AlarmPoint Systems AlarmPoint is an interactive alerting platform, designed to capture and enrich important events, to route those events to the correct person on any communication device, and to give that person the ability to solve, escalate, or enlist others to resolve the situation.

Through integrations, AlarmPoint can become the voice and interface of an automation engine or an intelligent application, such as HP Network Node Manager i Software. When NNMi detects an event that requires attention, AlarmPoint places phone calls or sends pages, instant messages, or e-mail messages to the appropriate personnel, vendors, or customers.

AlarmPoint is also persistent, escalating through multiple devices, communication mediums, and personnel until someone accepts responsibility or resolves the event. AlarmPoint gives the notified person instant, two-way communication with NNMi. Responses are processed immediately on the NNMi management server, enabling remote resolution of the event.

AlarmPoint Mobile Gateway, included with all AlarmPoint Enterprise licenses, extends the functionality of the AlarmPoint platform to provide mobile web access to critical applications. AlarmPoint Mobile Gateway provides access to enterprise applications from mobile devices for accessing and modifying ticket information, monitoring dashboards, and generating sophisticated reports.

AlarmPoint Express is included in bundled packages of NNMi. For information about purchasing other editions of AlarmPoint software with extended feature sets and AlarmPoint Mobile Gateway, contact your HP sales representative or sales@alarmpoint.com.

This chapter describes the available integrations:

- [HP NNMi–AlarmPoint Integration](#)
- [HP NNMi–AlarmPoint Mobile Gateway Integration](#)

HP NNMi–AlarmPoint Integration

About the HP NNMi–AlarmPoint Integration

By including AlarmPoint in the NNMi environment, network operations staff who use NNMi to monitor and manage their network devices gain intelligent alerting and two-way communication between personal communication tools and NNMi.

The HP NNMi–AlarmPoint integration supports event notifications (from NNMi to AlarmPoint) through the configuration of NNMi incident types. It also supports inbound actions (from AlarmPoint to NNMi) to acknowledge the original incident, alter its priority, and add informational annotations.

Each NNMi management server can be integrated with AlarmPoint, with AlarmPoint Mobile Gateway, or with both AlarmPoint and AlarmPoint Mobile Gateway.

Value

With the HP NNMi–AlarmPoint integration, the appropriate technician can be notified directly through voice, e-mail, pager, or another device. The event resolver receives information about the failure and is able to make decisions in real time, such as acknowledging, ignoring, annotating, or changing the priority of the event.

After the recipient selects a response on their remote device, AlarmPoint updates the NNMi incident in real time. The benefit is that this process is immediate—significantly faster than the time required for operations staff to notice the failures or malfunctions, determine who is on call, and then manually notify the correct person. Being able to take simple actions to update the incident from any device gives the event resolver a quick way to deal with many issues and communicate to other team members the current status of the incident.

During the process, AlarmPoint logs every notification, response, and action. In addition, AlarmPoint automatically annotates the original NNMi incident with status information.

The AlarmPoint product features a self-service web-based user interface for assigning responsible personnel to each job. AlarmPoint also includes an optional enhanced Subscription panel that allows both managed- and self-subscription to NNMi incidents.



AlarmPoint Express, a limited edition of AlarmPoint, is available for select versions of NNMi. AlarmPoint Express for HP NNMi has a reduced feature set and comes pre-configured for an NNMi integration. AlarmPoint Express is a great way to get started with the AlarmPoint product family. AlarmPoint Express is well-suited for a small production environment that does not require voice or distributed load capability.

Supported Versions

The information in this section applies to the following product versions:

- AlarmPoint version 4.0 or higher
- AlarmPoint Java Client version 4.0 or higher
- NNMi version 8.10 or higher



The NNMi integration enablement license is *not* required for NNMi integrations with any of the AlarmPoint products.

Documentation

The HP NNMi–AlarmPoint integration is fully described in the *AlarmPoint for HP Network Node Manager i-series Integration Guide*, which is included with the integration.

The AlarmPoint documentation suite describes the AlarmPoint features and capabilities in detail. The documentation suite is available for download from the AlarmPoint Customer Support Site at:

<https://connect.alarmpoint.com>



The *AlarmPoint Express for HP NNMi Quick Start Guide* provides an introduction to AlarmPoint's features. This guide describes how to install, configure, and maintain an HP NNMi–AlarmPoint Express integration.

Enabling the HP NNMi–AlarmPoint Integration

If you plan to implement both AlarmPoint integrations with NNMi, it is recommended that you enable the HP NNMi–AlarmPoint integration before you enable the HP NNMi–AlarmPoint Mobile Gateway integration.

The high-level steps for installing and configuring the AlarmPoint for NNMi integration are as follows. For detailed information, see the *AlarmPoint for HP Network Node Manager i-series Integration Guide*.

- 1 Install the AlarmPoint Java Client on the NNMi management server.
- 2 Install the NNMi-specific integration script for the AlarmPoint Java Client.
- 3 Install the Web Services Library on the AlarmPoint Webservers and the Application server.
- 4 *Optional.* Install the AlarmPoint subscription panel for NNMi on the AlarmPoint Webservers.
- 5 Install the AlarmPoint action scripts for NNMi by using the AlarmPoint Developer IDE.
- 6 Install the integration voice files to the AlarmPoint Application server.
- 7 Configure an Event Domain (and, optionally, a Subscription Domain) in AlarmPoint.
- 8 Configure a Web Services Client for NNMi.

- 9 Configure the NNMi incident types that should trigger the AlarmPoint scripts.
- 10 Validate that the integration can inject NNMi incident parameters for AlarmPoint notifications, and that AlarmPoint responses properly update the NNMi incident.

Using the HP NNMi–AlarmPoint Integration

NNMi and AlarmPoint interact by delivering notifications to users and injecting the responses back into NNMi. When NNMi detects a problem in the network (for example, a NonSNMPNodeUnresponsive incident), the following process occurs:

- 1 NNMi calls the AlarmPoint Client (APClient) with the parameters describing the problem (for example, the computer affected and the situation).
- 2 The APClient submits the information to the AlarmPoint Agent (APAgent).
- 3 The APAgent ensures delivery of the problem details to AlarmPoint, which in turn notifies the appropriate recipient.
- 4 The recipient responds to the notification by taking one of the actions listed in [Table 31](#). The recipient's acknowledgement, annotation, or priority change updates NNMi through a Web services call.

Table 31 Possible Responses to a Received Incident Notification

Action	Description
Acknowledge	User takes ownership of the incident, preventing further notifications to other users. The exception is subscription FYI notifications, which are reporting on the service outage. These notifications are not stopped until the problem has actually been solved.
Ignore	Stops notifying the current user.
Raise Priority	Increases the priority of the incident in NNMi by one level. (Voice only)
Lower Priority	Decreases the priority of the incident in NNMi by one level. (Voice only)
Set Priority Top	Sets the priority of the incident to Top. (Email, BES, and browser only)
Set Priority High	Sets the priority of the incident to High. (Email, BES, and browser only)
Set Priority Medium	Sets the priority of the incident to Medium. (Email, BES, and browser only)
Set Priority Low	Sets the priority of the incident to Low. (Email, BES, and browser only)
Annotate	Allows the user to append a message to the Notes field of the NNMi incident. (Non-HTML email only)

Disabling the HP NNMi–AlarmPoint Integration

To disable the integration, remove the components installed by the integration's executable archive.

For information about removing an AlarmPoint deployment, see the *AlarmPoint for HP Network Node Manager i-series Integration Guide*.

Troubleshooting the HP NNMi–AlarmPoint Integration

For information about optimizing and extending the integration, and any currently known issues, see the *AlarmPoint for HP Network Node Manager i-series Integration Guide*.

HP NNMi–AlarmPoint Mobile Gateway Integration

About the HP NNMi–AlarmPoint Mobile Gateway Integration

By including AlarmPoint Mobile Gateway in the NNMi environment, network operations staff can interact with NNMi incidents in a mobile device's web browser.

Each NNMi management server can be integrated with AlarmPoint, with AlarmPoint Mobile Gateway, or with both AlarmPoint and AlarmPoint Mobile Gateway.

Value

With the HP NNMi–AlarmPoint Mobile Gateway integration, an NNMi operator can use the web browser on a mobile device to interact in real time with NNMi incidents:

- Query NNMi for current incidents
- View the details of an incident
- Modify the properties, such as status, lifecycle state, or assigned NNMi operator, of an incident

Supported Versions

The information in this section applies to the following product versions:

- AlarmPoint Mobile Gateway version 4.0 or higher
- AlarmPoint Integration Agent version 4.0 or higher
- NNMi version 8.10 or higher



The NNMi integration enablement license is *not* required for NNMi integrations with any of the AlarmPoint products.

Documentation

The HP NNMi–AlarmPoint Mobile Gateway integration is fully described in the *AlarmPoint Mobile Gateway for HP Network Node Manager i-series Software* integration guide, which is included with the integration.

The AlarmPoint documentation suite describes the AlarmPoint features and capabilities in detail. The documentation suite is available for download from the AlarmPoint Customer Support Site at:

<https://connect.alarmpoint.com>

Enabling the HP NNMi–AlarmPoint Mobile Gateway Integration

If you plan to implement both AlarmPoint integrations with NNMi, it is recommended that you enable the HP NNMi–AlarmPoint integration before you enable the HP NNMi–AlarmPoint Mobile Gateway integration. See [Enabling the HP NNMi–AlarmPoint Integration](#) on page 383.

The high-level steps for installing and configuring the AlarmPoint Mobile Gateway for NNMi integration are as follows. For detailed information, see the *AlarmPoint Mobile Gateway for HP Network Node Manager i-series Software* integration guide.



If the HP NNMi–AlarmPoint integration was configured for an earlier version of NNMi, you can upgrade NNMi from version 8.1x to version 8.13 without affecting the integration configuration. Verify that the AlarmPoint version is 4.0 or higher.

- 1 Install the AlarmPoint Integration Agent on the AlarmPoint server.
- 2 Install the Web Services Library on the AlarmPoint Integration Agent and Webservers.
- 3 Install AlarmPoint Mobile Gateway on the AlarmPoint Webserver.
- 4 Install the NNMi Integration Service.
- 5 Create an NNMi user with the Web Service Client role.
- 6 Configure an Event Domain (if the HP NNMi–AlarmPoint integration is not already configured) and Integration Service in AlarmPoint.
- 7 Validate the interaction between AlarmPoint Mobile Gateway and NNMi.

Using the HP NNMi–AlarmPoint Mobile Gateway Integration

With the HP NNMi–AlarmPoint Mobile Gateway integration, you can perform the following actions on an incident in a mobile web browser:

- Add notes
- Update priority
- Update lifecycle state
- Update the Assigned To operator
- View most of the incident details (those that the AlarmPoint Mobile Gateway administrator has made available)
- Display the quick view of the Source Object and Node.

For more information about using the integration, see the *AlarmPoint Mobile Gateway for HP Network Node Manager i-series Software* integration guide and the *AlarmPoint Mobile Gateway Guide*.

Disabling the HP NNMi–AlarmPoint Mobile Gateway Integration

To disable the integration, remove the components installed by the integration's executable archive.

For information about removing an AlarmPoint Mobile Gateway deployment, see the *AlarmPoint Mobile Gateway for HP Network Node Manager i-series Software* integration guide.

Troubleshooting the HP NNMi–AlarmPoint Mobile Gateway Integration

For information about optimizing and extending the integration, and any currently known issues, see the *AlarmPoint Mobile Gateway for HP Network Node Manager i-series Software* integration guide.

CiscoWorks LAN Management Solution

Cisco Systems CiscoWorks LAN Management Solution (CiscoWorks LMS) is an integrated suite of management tools for configuring, administering, monitoring, and troubleshooting Cisco networks.

This chapter contains the following topics:

- [HP NNMi–CiscoWorks LMS Integration](#)
- [Enabling the HP NNMi–CiscoWorks LMS Integration](#)
- [Using the HP NNMi–CiscoWorks LMS Integration](#)
- [Changing the HP NNMi–CiscoWorks LMS Integration Configuration](#)
- [Disabling the HP NNMi–CiscoWorks LMS Integration](#)
- [Troubleshooting the HP NNMi–CiscoWorks LMS Integration](#)
- [HP NNMi–CiscoWorks LMS Integration Configuration Form Reference](#)

HP NNMi–CiscoWorks LMS Integration

The HP NNMi–CiscoWorks LMS integration provides actions for accessing CiscoWorks LMS tools from the NNMi console.

Value

The HP NNMi–CiscoWorks LMS integration adds CiscoWorks LMS information to NNMi, so that NNMi users can detect and investigate potential network problems for Cisco devices.

Supported Versions

The information in this chapter applies to the following product versions:

- CiscoWorks LMS version 3.1 or higher
- NNMi version 8.13 or higher

NNMi and CiscoWorks LMS must be installed on separate computers. The NNMi management server and the CiscoWorks LMS server computer can be of the same or different operating systems.

For the most recent information about supported hardware platforms and operating systems for NNMi, see the NNMi support matrix.

For the most recent information about supported hardware platforms and operating systems for CiscoWorks LMS, see the documentation for your version:

- CiscoWorks LMS version 3.1:
http://www.cisco.com/en/US/docs/net_mgmt/ciscoverks_lan_management_solution/3.1/install/guide/prereq.html
- CiscoWorks LMS version 3.2:
http://www.cisco.com/en/US/docs/net_mgmt/ciscoverks_lan_management_solution/3.2/install/guide1/prereq.html

Documentation

This chapter describes how to configure NNMi to communicate with CiscoWorks LMS and how to use the integration from the NNMi console.

Enabling the HP NNMi–CiscoWorks LMS Integration

On the NNMi management server, configure the connection between NNMi and CiscoWorks LMS by following these steps:

- 1 In the NNMi console, open the **HP NNMi–CiscoWorks LMS Integration Configuration** form (**Integration Module Configuration > CiscoWorks LMS**).
- 2 Select the **Enable Integration** check box to make the remaining fields on the form available.
- 3 Enter the information for connecting to the NNMi management server. For information about these fields, see [NNMi Management Server Connection](#) on page 393.
- 4 Enter the information for connecting to the CiscoWorks LMS server. For information about these fields, see [CiscoWorks LMS Server Connection](#) on page 394.
- 5 Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with connecting to the NNMi management server, click **Return**, and then adjust the values as suggested by the text of the error message.

- 6 Load the incident definitions for CiscoWorks LMS-managed devices:
 - a Change to the following directory:
 - *Windows*: %NnmInstallDir%\newconfig\HPOvNmsEvent
 - *UNIX*: \$NnmInstallDir/newconfig/HPOvNmsEvent
 - b Import the CiscoWorks LMS incident definitions by entering the following command:


```
nnmconfigimport.ovpl -f nnm-cisco-incidentConfig.xml \
          -u <username> -p <password>
```
- 7 Optional and recommended. Load the MIB definition files for the traps that CiscoWorks LMS-managed devices generate:
 - a Obtain the appropriate MIB files from the device media or the Cisco web site:


```
tools.cisco.com/Support/SNMP/do/SearchOID.do?local=en&step=1
```
 - b Change to the directory where the MIB files are stored.
 - c Use the `nnmloadmib.ovpl` command to load the appropriate MIB files for the managed environment. For example:


```
nnmloadmib.ovpl -load cpqghost.mib -u <username> -p <password>
```
 - d Verify that the MIBs loaded correctly, by entering the following command:


```
nnmloadmib.ovpl -list -u <username> -p <password>
```

Using the HP NNMi–CiscoWorks LMS Integration


The HP NNMi–CiscoWorks LMS integration provides links from the NNMi console to CiscoWorks LMS. The integration does not provide single sign-on between the products. You must enter your CiscoWorks LMS user credentials to view the CiscoWorks LMS pages.

Enabling the HP NNMi–CiscoWorks LMS integration adds the following actions to the NNMi console:

- **CiscoWorks Device Center**—Opens the CiscoWorks Device Center in the context of the selected node.
- **CiscoWorks CiscoView**—Opens CiscoWorks CiscoView in the context of the selected node.


Changing the HP NNMi–CiscoWorks LMS Integration Configuration

- 1 In the NNMi console, open the **HP NNMi–CiscoWorks LMS Integration Configuration** form (**Integration Module Configuration > CiscoWorks LMS**).
- 2 Modify the values as appropriate. For information about the fields on this form, see [HP NNMi–CiscoWorks LMS Integration Configuration Form Reference](#) on page 393.
- 3 Verify that the **Enable Integration** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.

 The changes take effect immediately. You do not need to restart `ovjboss`.

Disabling the HP NNMi–CiscoWorks LMS Integration

- 1 In the NNMi console, open the **HP NNMi–CiscoWorks LMS Integration Configuration** form (**Integration Module Configuration > CiscoWorks LMS**).
- 2 Clear the **Enable Integration** check box at the top of the form, and then click **Submit** at the bottom of the form. The integration actions are no longer available.

 The changes take effect immediately. You do not need to restart `ovjboss`.

Troubleshooting the HP NNMi–CiscoWorks LMS Integration

CiscoWorks LMS Actions Do Not Work

If you have verified the values in the **HP NNMi–CiscoWorks LMS Integration Configuration** form and you are still not able to open a CiscoWorks LMS page from the NNMi console, do the following:

- 1 Clear the web browser cache.
- 2 Clear all saved form or password data from the web browser.
- 3 Close the web browser window completely, and then re-open it.
- 4 Re-enter the values in the **HP NNMi–CiscoWorks LMS Integration Configuration** form.
- 5 Verify that CiscoWorks LMS is running.

OID Not Found in the MIB Cache Message in Traps

If the MIB definition files for the traps that CiscoWorks LMS-managed devices generate are not loaded in NNMi, you might see an error similar to the following text:

```
<Cia .1.3.6.1.4.1.11.5.7.5.2.1.1.1.7.0 with value 1 was not found within the mib cache>
```

To resolve these errors, load the MIBs as described in [step 7](#) on page 391.

HP NNMi–CiscoWorks LMS Integration Configuration Form Reference

The **HP NNMi–CiscoWorks LMS Integration Configuration** form contains the parameters for configuring communications between NNMi and CiscoWorks LMS. This form is available from the **Integration Module Configuration** workspace.



Only NNMi users with the Administrator role can access the **HP NNMi–CiscoWorks LMS Integration Configuration** form.

The **HP NNMi–CiscoWorks LMS Integration Configuration** form collects information for the following general areas:

- [NNMi Management Server Connection](#)
- [CiscoWorks LMS Server Connection](#)

To apply changes to the integration configuration, update the values on the **HP NNMi–CiscoWorks LMS Integration Configuration** form, and then click **Submit**.

NNMi Management Server Connection

[Table 32](#) lists the parameters for connecting to the NNMi management server. This is the same information that you use to open the NNMi console. You can determine many of these values by examining the URL that invokes an NNMi console session. Coordinate with the NNMi administrator to determine the appropriate values for this section of the configuration form.

Table 32 NNMi Management Server Information

Field	Description
NNMi SSL Enabled	The connection protocol specification. <ul style="list-style-type: none">• If the NNMi console is configured to use HTTPS, select the NNMi SSL Enabled check box. This is the default configuration.• If the NNMi console is configured to use HTTP, clear the NNMi SSL Enabled check box.
NNMi Host	The fully-qualified domain name of the NNMi management server. This field is pre-filled with host name that was used to access the NNMi console. Verify that this value is the name that is returned by the <code>nnmofficialfqdn.ovpl -t</code> command run on the NNMi management server.
NNMi Port	The port for connecting to the NNMi console. This field is pre-filled with the port that the jboss application server uses for communicating with the NNMi console, as specified in the following file: <ul style="list-style-type: none">• <i>Windows</i>: %NnmDataDir%\conf\nnm\props\nms-local.properties• <i>UNIX</i>: \$NnmDataDir/conf/nnm/props/nms-local.properties For non-SSL connections, use the value of <code>jboss.http.port</code> , which is 80 or 8004 by default (depending on the presence of another web server when NNMi was installed). For SSL connections, use the value of <code>jboss.https.port</code> , which is 443 by default.

Table 32 NNMi Management Server Information (cont'd)

Field	Description
NNMi User	The user name for connecting to the NNMi console. This user must have the NNMi Administrator or Web Service Client role.
NNMi Password	The password for the specified NNMi user.

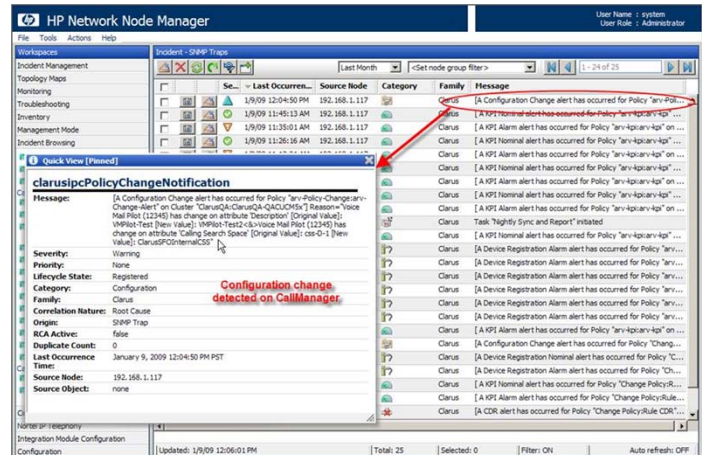
CiscoWorks LMS Server Connection

Table 33 lists the parameters for connecting to the CiscoWorks LMS server to open V pages. Coordinate with the CiscoWorks LMS administrator to determine the appropriate values for this section of the configuration.

Table 33 CiscoWorks LMS Management Server Information

CiscoWorks LMS Server Parameter	Description
CiscoWorks LMS SSL Enabled	The connection protocol specification for connecting to CiscoWorks LMS. <ul style="list-style-type: none"> If CiscoWorks LMS is configured to use HTTPS, select the CiscoWorks LMS SSL Enabled check box. This is the default configuration. If CiscoWorks LMS is configured to use HTTP, clear the CiscoWorks LMS SSL Enabled check box.
CiscoWorks LMS Host	The fully-qualified domain name of the CiscoWorks LMS server.
CiscoWorks LMS Port	The port for connecting to the CiscoWorks LMS web services. If you are using the default CiscoWorks LMS configuration, use port 1741 (for non-SSL connections to CiscoWorks LMS) or port 443 (for SSL connections to CiscoWorks LMS).

Clarus Systems ClarusIPC Plus⁺



Clarus Systems ClarusIPC Plus⁺ provides voice service testing; remote diagnostics of IP phone features; call detail record (CDR) based alerting and tracking; and reporting of configurations for Cisco Unified Communications Manager IP telephony systems during new deployments, upgrades, and ongoing operations.

Clarus Systems offers an integration of ClarusIPC Plus⁺ with HP Network Node Manager i Software.

HP offers an integration of ClarusIPC Plus⁺ with the NNM iSPI for IP Telephony. These integrations are mutually exclusive.

This chapter describes the available integrations:

- HP NNMi–Clarus Systems ClarusIPC Plus⁺ Integration
- HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ Integration

HP NNMi–Clarus Systems ClarusIPC Plus⁺ Integration

About the HP NNMi–Clarus Systems ClarusIPC Plus⁺ Integration

Clarus Systems provides and supports the HP NNMi–Clarus Systems ClarusIPC Plus⁺ integration. In this integration, ClarusIPC Plus⁺ forwards SNMP traps regarding IP telephony service test results, alerts based on set CDR policies, or alerts based on the Unified Communications Manager Configuration change policies to NNMi, which then generates incidents regarding the status of the IP telephony configuration and devices. NNMi provides a consolidated view of the entire network.

The integration provides for accessing several ClarusIPC Plus⁺ tools from these incidents in the NNMi console.

Value

The HP NNMi–Clarus Systems ClarusIPC Plus⁺ integration consolidates IP telephony device management by providing access from the NNMi console to the ClarusIPC Plus⁺ tools for IP telephony configuration change tracking and reporting.

Supported Versions

The information in this section applies to the following product versions:

- ClarusIPC Plus⁺ version 2.6.1 or higher
- NNMi version 8.10 or higher on the Windows operating system only

Documentation

The HP NNMi–Clarus Systems ClarusIPC Plus⁺ integration is fully described in the *ClarusIPC Plus⁺ HP NNMi Software Integration Guide*, which is included in the integration installation package.

The ClarusIPC Plus⁺ documentation suite contains additional documents that describe the ClarusIPC Plus⁺ features and capabilities in detail. The documentation suite is available for download from the Clarus Systems web site at:

www.support.clarussystems.com

Enabling the HP NNMi–Clarus Systems ClarusIPC Plus⁺ Integration

To obtain the HP NNMi–Clarus Systems ClarusIPC Plus⁺ integration installation package, contact Clarus Systems support.

For information about enabling the integration, see the *ClarusIPC Plus⁺ HP NNMi Software Integration Guide*, which is included in the integration installation package.

Using the HP NNMi–Clarus Systems ClarusIPC Plus⁺ Integration

Enabling the HP NNMi–Clarus Systems ClarusIPC Plus⁺ integration adds several URL actions to the NNMi console. For information about these URL actions, see the *ClarusIPC Plus⁺ HP NNMi Software Integration Guide*.



ClarusIPC Plus⁺ requires the use of the Microsoft Internet Explorer web browser. Open the NNMi console in Internet Explorer before launching a URL action that opens a ClarusIPC Plus⁺ window.

Disabling the HP NNMi–Clarus Systems ClarusIPC Plus⁺ Integration

For information about disabling the HP NNMi–Clarus Systems ClarusIPC Plus⁺ integration, contact Clarus Systems support.

Troubleshooting the HP NNMi–Clarus Systems ClarusIPC Plus⁺ Integration

For information about optimizing and extending the integration, and any currently known issues, see the *ClarusIPC Plus⁺ HP NNMi Software Integration Guide*.

HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ Integration

About the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ Integration

HP provides and supports the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ integration. With this integration, operators can access the ClarusIPC Plus⁺ features pertaining to IP telephony service tests and diagnostics; Cisco Unified Communications Manager Configuration change reports; and CDR monitoring policies. ClarusIPC Plus⁺ forwards SNMP traps regarding IP telephony service test results, alerts based on set CDR policies, or alerts based on the Unified Communications Manager Configuration change policies to NNMi, which then generates incidents regarding the status of the IP telephony configuration and devices. The NNM iSPI for IP Telephony provides the following:

- Workspaces and menus for launching to ClarusIPC Plus⁺ configuration change reports, various policies, test plans, and test results
- Launches to ClarusIPC Plus⁺ remote diagnostics tools for IP phones in the context of the selected IP phone
- Launches to ClarusIPC Plus⁺ test results, test details, and CDR policy details in the context of the alert incident selected in an NNMi incident view.

This integration provides access from the NNMi console to more ClarusIPC Plus⁺ tools than does the integration without the NNM iSPI for IP Telephony.

Value

The HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ integration adds advanced IP telephony service testing and diagnostics; CDR monitoring; and configuration change tracking and reporting to the NNM iSPI for IP Telephony.

Supported Versions

The information in this section applies to the following product versions:

- ClarusIPC Plus⁺ version 2.6.1 or higher
- NNMi version 8.11 or higher with an NNM iSPI Network Engineering Toolset Software license



NNMi version 8.11 is a patch to NNMi version 8.10.

- NNM iSPI for IP Telephony version 8.11 or higher, on any supported operating system



NNM iSPI for IP Telephony version 8.11 is a patch to the NNM iSPI for IP Telephony version 8.10.

Documentation

The HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ integration is fully described in the NNM iSPI for IP Telephony help, which is included with the iSPI.

The help (in PDF format) and additional NNM iSPI for IP Telephony documentation are available at:

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

Enabling the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ Integration

- 1 Prepare the NNMi management server:
 - a If the HP NNMi–Clarus Systems ClarusIPC Plus⁺ integration (provided by Clarus Systems) is installed on the NNMi management server, uninstall that integration before enabling the integration between the NNM iSPI for IP Telephony and ClarusIPC Plus⁺.

For information about how to uninstall the ClarusIPC Plus⁺ integration package, contact Clarus Systems support.

- b On the NNMi management server, install the following:
 - The most recent NNMi consolidated patch (if any)
 - The most recent NNM iSPI for IP Telephony consolidated patch (if any)

Patches are available at:

<http://h20230.www2.hp.com/selfsolve/patches>

- 2 On the NNMi management server, enable the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ integration as described in the NNM iSPI for IP Telephony help.

Using the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ Integration

Enabling the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ integration adds several workspaces, incident types, and URL actions to the NNMi console. For information about these URL actions, see the NNM iSPI for IP Telephony help.



ClarusIPC Plus⁺ requires the use of the Microsoft Internet Explorer web browser. Open the NNMi console in Internet Explorer before launching a URL action that opens a ClarusIPC Plus⁺ window.

Disabling the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ Integration

For information about disabling the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ integration, see the NNM iSPI for IP Telephony help.

Troubleshooting the HP NNM iSPI for IP Telephony–Clarus Systems ClarusIPC Plus⁺ Integration

For information about optimizing and extending the integration, and any currently known issues, see the NNM iSPI for IP Telephony help.

For help troubleshooting problems with ClarusIPC Plus⁺, contact Clarus Systems support.

HP Asset Manager

HP Asset Manager is an asset management platform for tracking and maintaining corporate inventory, including IT assets. Asset Manager provides the following functionality:

- Manage software license compliance, entitlements, and costs
- Discover, provision, manage, and improve the use of physical and virtual IT assets
- Use distributed sources of inventory, asset, and service data efficiently
- Automate management of equipment, space, cooling, and power

For information about purchasing Asset Manager, contact your HP sales representative.

This chapter contains the following topics:

- [HP NNMi–HP Asset Manager Integration](#)
- [Using the HP NNMi–HP Asset Manager Integration](#)

HP NNMi–HP Asset Manager Integration

When HP Network Node Manager i-suite Software is integrated with HP Asset Manager, the Asset Manager portfolio can be automatically populated with device information from the NNMi database.

Value

The HP NNMi–HP Asset Manager integration provides the following benefits:

- Reduces network traffic to the managed devices. NNMi discovers the devices, and Asset Manager synchronizes with the NNMi topology.
- Provides the complete network inventory to Asset Manager. This inventory is kept current through NNMi spiral discover.
- Simplifies network management configuration. Network discovery configuration is stored in NNMi only.

Supported Versions

The information in this chapter applies to the following product versions:

- Asset Manager version 5.2x with HP Connect-It version 4.1x
- NNMi version 9.00 or higher

NNMi and Asset Manager must be installed on separate computers. The NNMi management server and the Asset Manager computer can be of the same or different operating systems.

Documentation

The HP NNMi–HP Asset Manager integration is fully described in the *HP Connect-It Connectors* guide, which is available for the Connect-It and Integration Connectors product category at <http://h20230.www2.hp.com/selfsolve/manuals>.

Using the HP NNMi–HP Asset Manager Integration

The steps to enable the HP NNMi–HP Asset Manager integration take place on the Asset Manager server.

Enabling the HP NNMi–HP Asset Manager integration populates the Asset Manager portfolio with NNMi node, interface, and IP address information.

For information about enabling, using, disabling, and troubleshooting the HP NNMi–HP Asset Manager integration, see the *HP Connect-It Connectors* guide.

HP Business Availability Center My BSM

HP Business Availability Center (BAC) software provides tools for managing the availability of applications in production, monitoring system performance, monitoring infrastructure performance, and proactively resolving problems when they arise. BAC includes the My BSM portal for viewing reports and real-time product performance information. (Prior to BAC version 8.00, the portal is called My BAC.)

This chapter contains the following topics:

- [HP NNMi–HP BAC My BSM Integration](#)
- [Default NNMi Modules for My BSM](#)
- [Configuring the Demonstration Portlets](#)
- [Creating Custom NNMi Portlets](#)
- [Configuring Single Sign-On for the HP NNMi–HP BAC My BSM Integration](#)
- [Troubleshooting the HP NNMi–HP BAC My BSM Integration](#)
- [HP NNMi–HP BAC My BSM Configuration Form Reference](#)

HP NNMi–HP BAC My BSM Integration

The HP NNMi–HP BAC My BSM integration is an enablement for viewing NNMi in the My BSM portal. The integration provides templates for adding NNMi and NNM iSPI for Performance portlets to a My BSM portal. For a quick portal demonstration, you can use the configuration tool in the NNMi console to customize these templates for your environment.

The My BSM administrator can further customize the configuration and the access rights of the default portlets by using the standard My BSM administration interface. The My BSM administrator can also use the My BSM administration interface to create custom portlets for other views of NNMi and the NNMi Smart Plug-ins (NNM iSPIs), and to combine views from multiple NNMi management servers on one portal page.

Value

The HP NNMi–HP BAC My BSM integration extends the information available through the My BSM portal.

Supported Versions

The information in this chapter applies to the following product versions:

- BAC version 7.54, 8.03, or 8.04



To integrate with BAC version 9.x, use the BAC component gallery.

- NNMi version 8.11 or higher
- *Optional.* NNM iSPI for Performance version 8.1x

For the most recent information about supported hardware platforms and operating systems, see the support matrices for both products.

Documentation

This chapter describes how to configure the default NNMi portlets for My BSM from the NNMi console.

The *Using My BAC Guide*, which is included on the BAC 7.x product media, describes how to configure and maintain My BAC.

The *Using My BSM Guide*, which is included on the BAC 8.x product media, describes how to configure and maintain My BSM.

Default NNMi Modules for My BSM

NNMi provides the following My BSM modules for configuration in the NNMi console:

- The NNMi demonstration module is defined in the `NOC_Demo_Portal.xml` template file. This module presents some key network status information as described in [Table 34](#).
- The NNMi and NNM iSPI for Performance demonstration module is defined in the `NOC_Demo_Portal_iSPIPerf.xml` template file. This module adds some NNM iSPI for Performance reports to the network status information of the NNMi demonstration module. [Table 35](#) on page 405 describes this module.

Table 34 NNMi Demonstration Module Contents

Page	Portlet	Portlet Description
Overview Map	Key Operations Map	The NNMi topology map for the specified node group.
Network Status/Device Health	Node Group Status	Equivalent to the results of the Actions > Status Details menu command for the specified node group in the NNMi console.
	Network Status	Equivalent to the Node Groups inventory view in the NNMi console.

Table 35 NNMi and NNM iSPI for Performance Demonstration Module Contents

Page	Portlet	Portlet Description
Overview Map	Key Operations Map	The NNMi topology map for the specified node group.
Network Status/Device Health	Node Group Status	Equivalent to the results of the Actions > Status Details menu command for the specified node group in the NNMi console.
	Network Status	Equivalent to the Node Groups inventory view in the NNMi console.
	Top-N Memory Utilization	The live NNM iSPI for Performance component health report showing the top 10 nodes by memory utilization.
	Top-N CPU Utilization	The live NNM iSPI for Performance component health report showing the top 10 nodes by CPU utilization.
NNM iSPI for Performance Exceptions	Top-N Devices by Component Exceptions	The live NNM iSPI for Performance component health dashboard showing the top 5 nodes by CPU utilization exceptions and the top 5 nodes by memory utilization exceptions.

Configuring the Demonstration Portlets

This section describes the initial configuration of the demonstration My BSM modules. The My BSM administrator can completely customize portlet content and user access to the portlets.

- 1 On the NNMi management server, create the module configuration XML file:
 - a In the NNMi console, open the **HP NNMi–HP BAC My BSM Configuration** form (**Integration Module Configuration > HP BAC My BSM**).
 - b Select one of the named XML files to customize:
 - If the NNM iSPI for Performance is not installed in your environment, select the `NOC_Demo_Portal.xml` file.
 - If the NNM iSPI for Performance is installed in your environment, select the `NOC_Demo_Portal_iSPIPerf.xml` file.
 - c Click **Load**.
 - d On each page of the **HP NNMi–HP BAC My BSM Configuration** form, edit the supplied text as appropriate, and then click **Next**. For information about these fields, see [HP NNMi–HP BAC My BSM Configuration Form Reference](#) on page 412.
 - e After navigating through all pages of the **HP NNMi–HP BAC My BSM Configuration** form, click **Finish**, and then save the XML file to a known place on your computer.
 - f Close the **HP NNMi–HP BAC My BSM Configuration** form.
- 2 Import the module configuration into BAC:
 - a On the **BAC Administration** tab, under **My BSM** (or **My BAC**), click **Import portlets and modules** (one of the options for managing portlet definitions).
 - b On the **Import My BSM Objects** page (or the **Import My BAC Objects** page), click **Browse**, and then select the XML file that you saved from the NNMi console.
 - c Select the **Replace same Portlet Definitions** check box.
 - d Select the **Replace same Modules** check box.
 - e Click **Import**.

The **Import Status** window displays the results of the operation. If the import did not succeed, verify that both check boxes are selected, and then retry the import.
- 3 View the module in My BSM or My BAC:
 - Verify that each portlet displays the expected information.
 - Use the My BSM or My BAC administration tools to define which users can access the new portlets, to reorganize the pages, to edit the portlet definitions, and so forth.

Creating Custom NNMi Portlets

The easiest way to create a new portlet that displays NNMi or iSPI information is as follows:

- 1 In the My BSM administration interface, copy an existing NNMi portlet definition.
- 2 Change the URL in the new portlet definition to point to the information that you want to display in the portal.

As you edit the portlet definitions, follow the HTML code structure used in the demonstration portlets. For a description of the HTML code structure, see [Portlet Definition HTML Reference](#) on page 408.

Determining the Portlet URL

URL for an NNMi console window

For information about how to create a URL for launching an NNMi console window directly, see **Help > NNMi Documentation Library > Integrate NNMi Elsewhere with URLs** in the NNMi console.

URL for an NNM iSPI for Performance report

The procedure for determining the URL for launching an NNM iSPI for Performance report depends on which web browser you are using to view the report.



The URL for accessing the report is the same in any web browser.

In Mozilla Firefox, to determine the URL for launching an NNM iSPI for Performance report, follow these steps:

- 1 Run the NNM iSPI for Performance report.
- 2 *Optional.* Click **Show Options** at the top of the report, and then customize the report view.
- 3 Click **Show URL** at the top of the report.

The URL for the report is visible below the report banner and customization links. You can copy the URL for use in a portlet definition.

- 4 *Optional.* Click **Hide URL** to hide the report URL from view.

In Microsoft Internet Explorer, to determine the URL for launching an NNM iSPI for Performance report, follow these steps:

- 1 Run the NNM iSPI for Performance report.
- 2 *Optional.* Click **Show Options** at the top of the report, and then customize the report view.
- 3 Click **Add Bookmark** at the top of the report.
- 4 In the **Add a Favorite** window, click **Add**.
- 5 In the favorites list, right-click the favorite that you created in [step 4](#), and then click **Properties**.

The **URL** field displays the URL for the report. You can copy the URL from this field for use in a portlet definition.

Portlet Definition HTML Reference

The following rules apply to the HTMLPortlet type in BAC:

- Use a single quote character (') instead of the standard double quote character (").
- For every iframe start tag (<iframe>) use a corresponding iframe end tag (</iframe>) because some web browsers do not recognize empty-element <iframe/> tags correctly.
- In the iframe definition, specify one of the following values for id:

id Value	Description
nnmi-portlet	The ID of an iframe that displays an NNMi URL. This ID provides consistent configuration for human readers.
nnmi-auth	The ID of an iframe that manages single sign-on to the NNM iSPI for Performance. The JavaScript function for loading the NNM iSPI for Performance portlet interprets this ID value.
ispiperf-portlet	The ID of an iframe that displays an NNM iSPI for Performance report URL. The JavaScript function for loading the NNM iSPI for Performance portlet interprets this ID value.

NNMi portlet HTML structure

A BAC portlet that displays an NNMi URL contains a single iframe element that identifies the NNMi URL to display. The structure is as follows:

```
<html>
<head></head>
<body>
<iframe id='nnmi-portlet' src='<NNMi_URL>'></iframe>
</body>
</html>
```

Replace *<NNMi_URL>* with the URL for launching an NNMi console window.

For example, the following code defines a portlet that displays the status for the Routers node group. In this example, the height and width values are the suggested values for this portlet. You can change these values as needed.

```
<html>
<head></head>
<body>
<iframe id='nnmi-portlet'
  src='http://nnmi.example.com:8004/nnm/launch
      ?cmd=runTool
      &tool=nodegroupstatus
      &nodegroup=Routers
      &menus=false'
  width='100%'
  height='425px'>
</iframe>
</body>
</html>
```


NNM iSPI for
Performance portlet
HTML structure

A BAC portlet that displays an NNM iSPI for Performance report URL contains the following elements:

- Within the portlet header, the declaration of the `loadIspiPerf()` JavaScript function. This declaration defines the NNM iSPI for Performance report to display.
- One `iframe` element that handles single-sign on from NNMi to the NNM iSPI for Performance.
- A second `iframe` element that displays the NNM iSPI for Performance report named in the `loadIspiPerf()` function declaration of the portlet header.

The structure of a BAC portlet that displays an NNM iSPI for Performance report URL is as follows:

```
<html>
<head>
<script id='ispiperf-load'>function loadIspiPerf()
{
  var ispiperf_url='<Report_URL>';
  document.getElementById('ispiperf-portlet').src =
    ispiperf_url;
}
</script>
</head>
<body onload='loadIspiPerf();'>
<iframe id='nnmi-auth'
  src='http:<NNMi_host>:<NNMi_port>/nnm/launch?cmd=isRunning'>
</iframe>
<iframe id='ispiperf-portlet'></iframe>
</body>
</html>
```

Replace `<Report_URL>` with the URL for launching an NNM iSPI for Performance report. Replace `<NNMi_host>` and `<NNMi_port>` with the fully-qualified domain name of the NNMi management server and the port number for accessing NNMi, respectively.

For example, the following code defines a portlet that displays the node health report for the top N nodes. In this example, the height and width values are the suggested values for this portlet. You can change these values as needed.

```
<html>
<head>
<script id='ispiperf-load'>function loadIspiPerf()
{
  var ispiperf_url=
    'http://nnmi.example.com:8004/ssoservlet/protected
    /reports
    ?reportURL=http://ispiperf.example.com:9300/PerfSpi
    /PerfSpi
    ?package=NodeHealth
    &report=Top%20N%20Live
    &element=All%20Nodes/Components&timeperiod=
    &dow=
    &hod=
    &metric=CPU%20Utilization%20(Avg%25)
    &namespaceID=ErsAuthenticationProvider
    &ssodomain=example.com';
```

```
        document.getElementById('ispiperf-portlet').src =
            ispiperf_url;
    }
</script>
</head>
<body onload='loadIspiPerf();'>
<iframe id='nnmi-auth'
    src='http://nnmi.example.com:8004/nm/launch
        ?cmd=isRunning'
    width='100%'
    height='1px'>
</iframe>
<iframe id='ispiperf-portlet'
    width='100%'
    height='750px'>
</iframe>
</body>
</html>
```

Configuring Single Sign-On for the HP NNMi–HP BAC My BSM Integration

Single sign-on is available for all HP enterprise applications that use identical initialization string values and also share a common network domain name. For information about configuring single sign-on for BAC My BSM, see [Configuring Single Sign-On for the HP NNMi–HP BAC My BSM Integration](#) on page 115.

Troubleshooting the HP NNMi–HP BAC My BSM Integration

The NNMi Portlets Appear As a Sign-in Page

Verify the single sign-on configuration:

- 1 Sign in to the NNMi console with the user name for logging on to My BSM.
If sign in is not successful, ask the NNMi administrator to configure an account for the My BSM user.
- 2 Make sure that BAC and NNMi use the same initialization strings as described in [Configuring Single Sign-On for the HP NNMi–HP BAC My BSM Integration](#) on page 410.

Also see [Single Sign-On Does Not Work Correctly](#) on page 411.

An NNMi Portlet Does Not Load Correctly

In the My BSM administration interface, verify the NNMi management server host name and the port number in the portlet URL.

An NNM iSPI for Performance Portlet Does Not Load Correctly

In the My BSM administration interface, verify the NNM iSPI for Performance server host name, port number, and single-sign on domain in the portlet URL.

An NNM iSPI for Performance Portlet Displays an AsynchWait_Requests Error

As a BAC portal page loads an NNM iSPI for Performance portlet, it requests information from the NNM iSPI for Performance Cognos database. When a page contains multiple NNM iSPI for Performance portlets, simultaneous requests to the Cognos database from the single web browser session can result in an `AsynchWait_Requests` error. Reload the portal page.

Single Sign-On Does Not Work Correctly

All NNMi and NNM iSPI for Performance portlets do not load. The web browser might close with a message similar to the following:

The page you requested cannot be displayed because the LW-SSO host has logged out.

Verify that all application servers that participate in a single sign-on integration are set to the same GMT time with a maximum difference of 15 minutes.

My BSM Reports HTML Validation Errors When I Save A Portlet Definition

Before you can save new portlets in My BSM, you must configure a setting in a BAC configuration file.

Edit the <HP Business Availability Center root directory>\HPBAC\conf\dashboard.properties file to add the following line:

```
Block-URL-Injections=false
```

HP NNMi—HP BAC My BSM Configuration Form Reference

Table 36 lists the fields included on the pages of the **HP NNMi—HP BAC My BSM Configuration** form. Coordinate with the NNM iSPI for Performance administrator to determine the correct value for the NNM iSPI for Performance fields. Text fields may contain any characters. NNMi does not validate configured values. Verify the new portlets in the My BSM portal.

Table 36 Module Configuration Information

Field	Description
Name	The name of the portal module. This text assists navigation within the My BSM portal.
Description	Text that describes the portal module. This text is visible in the My BSM administration interface.
Page Title	The name of a portal page. This text assists navigation within the portal.
Portlet Title	The name of the portlet as it appears in the portal.
Portlet Type	A field required for the My BSM configuration. This field is currently read-only.
NNMi Machine	The URL for accessing the NNMi management server. This field is pre-filled with the host name and the port number for connecting to the NNMi management for the current NNMi console session. <ul style="list-style-type: none">• For the portlet to access the default NNMi management server, leave the default setting.• For the portlet to access a different NNMi management server, manually enter the correct URL.
NNMi Nodegroup	A list of node groups on the NNMi management server for the current NNMi console session. <ul style="list-style-type: none">• If the portlet accesses the default NNMi management server, select a node group from the list.• If the portlet accesses a different NNMi management server, manually enter the correct node group name.
iSPI for Performance Machine	The URL for accessing the NNM iSPI for Performance server. This field is pre-filled with the host name and the port number for connecting to the NNM iSPI for Performance server from the current NNMi console session. <ul style="list-style-type: none">• For the portlet to access the default NNM iSPI for Performance server, leave the default setting.• For the portlet to access a different NNM iSPI for Performance server, manually enter the correct URL.

Table 36 Module Configuration Information (cont'd)

Field	Description
SSO Domain	The domain for single sign-on to the NNM iSPI for Performance.
Disable iSPI for Performance portlets check box	<p>The Disable iSPI for Performance portlets check box is available on configuration form pages that define portlet access to NNM iSPI for Performance.</p> <p>If this check box is selected when the page first appears, the NNM iSPI for Performance is not configured on this NNMi management server. Therefore, the fields related to the NNM iSPI for Performance are unset.</p> <p>Clear this check box to enable the fields so that you can manually enter the information for accessing the NNM iSPI for Performance.</p>

HP Business Service Management Topology

HP Business Service Management (BSM) software provides tools for managing the availability of applications in production, monitoring system performance, monitoring infrastructure performance, and proactively resolving problems when they arise.

For information about purchasing BSM, contact your HP sales representative.

This chapter contains the following topics:

- [HP NNMi–HP BSM Topology Integration](#)
- [Enabling the HP NNMi–HP BSM Topology Integration](#)
- [Using the HP NNMi–HP BSM Topology Integration](#)
- [Changing the HP NNMi–HP BSM Topology Integration Configuration](#)
- [Disabling the HP NNMi–HP BSM Topology Integration](#)
- [Troubleshooting the HP NNMi–HP BSM Topology Integration](#)
- [HP NNMi–HP BSM Topology Integration Configuration Form Reference](#)

HP NNMi–HP BSM Topology Integration

The HP NNMi–HP BSM Topology integration populates the BSM operational database (ODB) with the NNMi topology. BSM stores each device in the NNMi topology as a configuration item (CI). BSM users and integrated applications can see the relationships among network devices.

Additionally, the integration stores the identifier of populated CIs in the NNMi database. Uses for the CIs of the NNMi-managed devices include the following:

- The HP NNMi–HP BSM Operations Management integration can associate incidents regarding NNMi-managed devices with BSM CIs. For more information, see [Configuration Item Identifiers](#) on page 476.
- The agent implementation of the HP NNMi–HPOM integration can associate incidents regarding NNMi-managed devices with BSM CIs. For more information, see [Configuration Item Identifiers](#) on page 494.



NNMi cannot simultaneously integrate with HP BSM topology and HP Universal CMDB (UCMDB). If you want NNMi information in both databases, do *both* of the following in any order:

- Configure the HP NNMi–HP BSM Topology integration, as described in this chapter.
- Configure the BSM integration with UCMDB, as described in the *UCMDB Data Flow Management Guide*.

Value

The HP NNMi–HP BSM Topology integration sets up NNMi as the authoritative source for network device status and relationship information.

Supported Versions

The information in this chapter applies to the following product versions:

- BSM version 9.00 or higher
- NNMi version 9.00 or higher

NNMi and BSM must be installed on separate computers. The NNMi management server and the BSM gateway server computer can be of the same or different operating systems.

For the most recent information about supported hardware platforms and operating systems, see the support matrices for both products.

Documentation

This chapter describes how to configure NNMi to communicate with BSM.

The BSM documentation suite describes the BSM features and capabilities in detail. The documentation suite is included on the BSM product media.

Enabling the HP NNMi–HP BSM Topology Integration



NNMi cannot simultaneously integrate with HP BSM topology and HP UCMDB. If the HP NNMi–HP UCMDB integration is configured on this NNMi management server, disable that configuration before enabling the HP NNMi–HP BSM Topology integration.

On the NNMi management server, configure the connection between NNMi and BSM by following these steps:

- 1 In the NNMi console, open the **HP NNMi–HP BSM Topology Integration Configuration** form (**Integration Module Configuration > HP BSM Topology**).
- 2 Select the **Enable Integration** check box to make the remaining fields on the form available.

- 3 Enter the information for connecting to the NNMi management server. For information about these fields, see [NNMi Management Server Connection](#) on page 419.
- 4 Enter the information for connecting to the BSM gateway server. For information about these fields, see [BSM Gateway Server Connection](#) on page 420.
- 5 *Optional.* Enter the information that describes which NNMi nodes should be maintained in BSM. For information about these fields, see [BSM Topology Filter](#) on page 421.
- 6 Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with connecting to the NNMi management server, click **Return**, and then adjust the values as suggested by the text of the error message.

Using the HP NNMi–HP BSM Topology Integration

The HP NNMi–HP BSM Topology integration populates the following CI types in the BSM ODB:

- InfrastructureElement > Node
The nodes in the NNMi topology. You can limit the set of nodes as described in [BSM Topology Filter](#) on page 421.
- InfrastructureElement > NodeElement> HardwareBoard
The cards associated with the Node CIs that the integration populates in BSM.
- InfrastructureElement > NodeElement> Interface
The interfaces associated with the Node CIs that the integration populates in BSM.
- InfrastructureElement > NodeElement> PhysicalPort
The ports associated with the Node CIs that the integration populates in BSM.
- InfrastructureElement > NetworkEntity > IPAddress
The IP addresses of the interfaces associated with the Node CIs that the integration populates in BSM.
- InfrastructureElement > NetworkEntity > IpSubnet
All subnets in the NNMi topology.
- InfrastructureElement > NetworkEntity > Layer2Connection
The NNMi Layer 2 connections with at least two connection ends that the integration populates as Node CIs in BSM.

For each CI created in the BSM ODB, the integration stores the ODB identifier in the NNMi database.




By default, NNMi does not discover end nodes. Update the NNMi discovery and monitoring configuration to include the end nodes that you want to see in BSM.

The HP NNMi–HP BSM Topology integration forwards NNMi information and updates to the BSM ODB as a one-way communication. Because NNMi does not know or control how the BSM CI information is used, the integration relies on the BSM aging policies to delete CIs that have not been updated for a set period of time.

The HP NNMi–HP BSM Topology integration enables other products to use the NNMi topology information when they integrate with BSM. There is no direct user interaction with this integration.


Changing the HP NNMi–HP BSM Topology Integration Configuration

- 1 In the NNMi console, open the **HP NNMi–HP BSM Topology Integration Configuration** form (**Integration Module Configuration > HP BSM Topology**).
- 2 Modify the values as appropriate. For information about the fields on this form, see [HP NNMi–HP BSM Topology Integration Configuration Form Reference](#) on page 419.
- 3 Verify that the **Enable Integration** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.

 The changes take effect immediately. You do not need to restart `ovjboss`.

Disabling the HP NNMi–HP BSM Topology Integration

- 1 In the NNMi console, open the **HP NNMi–HP BSM Topology Integration Configuration** form (**Integration Module Configuration > HP BSM Topology**).
- 2 Clear the **Enable Integration** check box at the top of the form, and then click **Submit** at the bottom of the form. The integration URL actions are no longer available.

 The changes take effect immediately. You do not need to restart `ovjboss`.

Troubleshooting the HP NNMi–HP BSM Topology Integration

For information about troubleshooting the connection to the ODB, see the BSM documentation suite.

HP NNMi–HP BSM Topology Integration Configuration Form Reference

The **HP NNMi–HP BSM Topology Integration Configuration** form contains the parameters for configuring communications between NNMi and BSM. This form is available from the **Integration Module Configuration** workspace.



Only NNMi users with the Administrator role can access the **HP NNMi–HP BSM Topology Integration Configuration** form.

The **HP NNMi–HP BSM Topology Integration Configuration** form collects information for the following areas:

- [NNMi Management Server Connection](#) on page 419
- [BSM Gateway Server Connection](#) on page 420
- [BSM Topology Filter](#) on page 421

To apply changes to the integration configuration, update the values on the **HP NNMi–HP BSM Topology Integration Configuration** form, and then click **Submit**.

NNMi Management Server Connection

Table 37 lists the parameters for connecting to the NNMi management server. This is the same information that you use to open the NNMi console. You can determine many of these values by examining the URL that invokes an NNMi console session. Coordinate with the NNMi administrator to determine the appropriate values for this section of the configuration form.

Table 37 NNMi Management Server Information

Field	Description
NNMi SSL Enabled	The connection protocol specification. <ul style="list-style-type: none"> • If the NNMi console is configured to use HTTPS, select the NNMi SSL Enabled check box. This is the default configuration. • If the NNMi console is configured to use HTTP, clear the NNMi SSL Enabled check box.
NNMi Host	The fully-qualified domain name of the NNMi management server. This field is pre-filled with host name that was used to access the NNMi console. Verify that this value is the name that is returned by the <code>nnmofficialfqdn.ovpl -t</code> command run on the NNMi management server.
NNMi Port	The port for connecting to the NNMi console. This field is pre-filled with the port that the jboss application server uses for communicating with the NNMi console, as specified in the following file: <ul style="list-style-type: none"> • <i>Windows:</i> %NnmDataDir%\conf\nnm\props\nms-local.properties • <i>UNIX:</i> \$NnmDataDir/conf/nnm/props/nms-local.properties For non-SSL connections, use the value of <code>jboss.http.port</code> , which is 80 or 8004 by default (depending on the presence of another web server when NNMi was installed). For SSL connections, use the value of <code>jboss.https.port</code> , which is 443 by default.

Table 37 NNMi Management Server Information (cont'd)

Field	Description
NNMi User	The user name for connecting to the NNMi console. This user must have the NNMi Administrator or Web Service Client role.
NNMi Password	The password for the specified NNMi user.

BSM Gateway Server Connection

Table 38 lists the parameters for connecting to the BSM gateway server to communicate with the BSM topology database. Coordinate with the BSM administrator to determine the appropriate values for this section of the configuration.

Table 38 BSM Gateway Server Information

BSM Gateway Server Parameter	Description
BSM SSL Enabled	The connection protocol specification for connecting to BSM. <ul style="list-style-type: none"> If BSM is configured to use HTTPS, select the BSM SSL Enabled check box. This is the default configuration. If BSM is configured to use HTTP, clear the BSM SSL Enabled check box.
BSM Host	The fully-qualified domain name of the BSM gateway server.
BSM Port	The port for connecting to BSM. If you are using the default BSM configuration, use port 80 (for non-SSL connections to BSM).
BSM User	The user name for connecting to the BSM topology database. The value must be <code>admin</code> .
BSM Password	The password for the specified BSM user. The value must be <code>admin</code> .

BSM Topology Filter

By default, the HP NNMi–HP BSM Topology integration conveys information about all nodes and interfaces in the NNMi topology to BSM. If you want the integration to maintain only a subset of the NNMi topology information in BSM, specify one or both of the optional node groups as described in this section.

The two scenarios for the filtering NNMi topology information are as follows:

- **Definitive**—In NNMi, create one node group that explicitly defines every NNMi node to be included in the BSM topology. This approach requires an intimate knowledge of your network topology.

For example, you might create a node group called `BSM_Topology` containing the following types of devices:

- The application servers in the managed environment
- The routers and switches that connect the application servers

In this case, specify the node group (for example, `BSM_Topology`) as the topology filter node group. Do not specify an additional connections node group.

The integration forwards information about every node in the specified topology filter node group (for example, `BSM_Topology`) and ignores all other nodes in the NNMi topology.

- **Additive**—In NNMi, identify (or create) a node group that defines the core infrastructure of the monitored network, and then create another node group that defines the end nodes of interest.

For example, you might create the following NNMi node groups:

- The `BSM_Core` group that contains the Networking Infrastructure Devices node group and other key connective devices
- The `BSM_End_Nodes` group that contains the application servers in the managed network

In this case, specify the first node group (for example, `BSM_Core`) as the topology filter node group. Also, specify the second node group (for example, `BSM_End_Nodes`) as the additional connections node group.

The integration forwards information about every node in the topology filter node group (for example, `BSM_Core`). The integration then examines each node in the additional connections node group (for example, `BSM_End_Nodes`) as follows:

- If the node is connected to one or more nodes in the topology filter node group, the integration forwards the information about that node to BSM.
- If the node is not connected to any of the nodes in the topology filter node group, the integration ignores that node.

Table 39 lists the optional parameters for specifying a BSM topology filter and provides information about entering values for these parameters.

Table 39 BSM Topology Filter Information

BSM Topology Filter Parameter	Description
Topology Filter Node Group	<p>The NNMi node group containing the primary set of nodes to populate in BSM. The integration populates the ODB with information about every node in this node group.</p> <p>Enter the name of the node group exactly as it is written (with no quotation marks or extra characters) in the Name field of the Node Group form in NNMi.</p> <p>If you do not specify a topology filter node group, the HP NNMi–HP BSM Topology integration populates the ODB with all nodes and interfaces in the NNMi topology. In this case, the integration ignores the value of the Additional Connections Node Group field.</p>
Additional Connections Node Group	<p>The NNMi node group containing hints of additional nodes to populate in BSM. The integration populates the ODB with information about only those nodes in this node group that are connected (in the NNMi topology) to one or more nodes in the topology filter node group.</p> <p>Enter the name of the node group exactly as it is written (with no quotation marks or extra characters) in the Name field of the Node Group form in NNMi.</p> <p>If you specify a topology filter node group and specify an additional connections node group, the HP NNMi–HP BSM Topology integration forwards information about the nodes and interfaces in the topology filter node group and the connected nodes in the additional connections node group.</p> <p>If you specify a topology filter node group but do not specify an additional connections node group, the HP NNMi–HP BSM Topology integration forwards information about the nodes and interfaces in the topology filter node group only.</p> <p>If you do not specify a topology filter node group, the HP NNMi–HP BSM Topology integration populates the ODB with all nodes and interfaces in the NNMi topology. In this case, the integration ignores the value of the Additional Connections Node Group field.</p>

HP Universal CMDB

HP Universal CMDB (UCMDB) automatically maintains accurate, up-to-date information on infrastructure and application relationships through native integration to HP Discovery and Dependency mapping (DDM). UCMDB is beneficial for the following tasks:

- Using impact modeling to show the rippling effect of infrastructure and application changes before they occur.
- Tracking actual planned and unplanned changes through discovered change history.
- Gaining a shared, authoritative view of the environment through awareness of existing data repositories.

For information about purchasing UCMDB, contact your HP sales representative.

This chapter contains the following topics:

- [HP NNMi–HP UCMDB Integration](#)
- [Using the HP NNMi–HP UCMDB Integration](#)

HP NNMi–HP UCMDB Integration

The HP NNMi–HP UCMDB integration shares NNMi topology information with UCMDB. UCMDB stores each device in the NNMi topology as a configuration item (CI). UCMDB applies Discovery and Dependency Mapping (DDM) patterns to the CIs for the NNMi topology to predict the impact of a device failure. This impact analysis is available from the UCMDB user interface and also from the NNMi console.

Additionally, the integration stores the identifier of populated CIs in the NNMi database. Uses for the CIs of the NNMi-managed devices include the following:

- The HP NNMi–HP BSM Operations Management integration can associate incidents regarding NNMi-managed devices with UCMDB CIs. For more information, see [Configuration Item Identifiers](#) on page 476.
- The agent implementation of the HP NNMi–HPOM integration can associate incidents regarding NNMi-managed devices with UCMDB CIs. For more information, see [Configuration Item Identifiers](#) on page 494.



NNMi cannot simultaneously integrate with HP BSM topology and HP Universal CMDB (UCMDB). If you want NNMi information in both databases, do *both* of the following in any order:

- Configure the HP NNMi–HP BSM Topology integration, as described in [HP Business Service Management Topology](#) on page 415.
- Configure the BSM integration with UCMDB, as described in the *UCMDB Data Flow Management Guide*.

Value

The HP NNMi–HP UCMDB integration sets up NNMi as the authoritative source for network device relationships.

Supported Versions

The information in this chapter applies to the following product versions:

- UCMDB version 8.03 or higher
- NNMi version 9.00 or higher

NNMi and UCMDB cannot be installed on the same computer. The two products must be installed on different computers in either of the following configurations:

- Different operating systems. For example, the NNMi management server is a Linux system, and the UCMDB server is a Windows system.
- The same operating system. For example, the NNMi management server is a Windows system, and the UCMDB server is a second Windows system.

For the most recent information about supported hardware platforms and operating systems, see the support matrices for both products.

Documentation

The HP NNMi–HP UCMDB integration is fully described in the *HP Universal CMDB–HP Network Node Manager (NNMi) Integration Guide*, which is included on the UCMDB product media.

Using the HP NNMi–HP UCMDB Integration



NNMi cannot simultaneously integrate with HP BSM topology and HP Universal CMDB. If the HP NNMi–HP BSM Topology integration is configured on this NNMi management server, disable that configuration before enabling the HP NNMi–HP UCMDB integration.

For information about enabling, using, disabling, and troubleshooting the HP NNMi–HP UCMDB integration, see the *HP Universal CMDB–HP Network Node Manager (NNMi) Integration Guide*.

HP Network Automation

HP Network Automation software (NA) tracks, regulates, and automates configuration and software changes across globally distributed, multi-vendor networks through process-powered automation.

This chapter contains the following topics:

- [HP NNMi–HP NA Integration](#)
- [Enabling the HP NNMi–HP NA Integration](#)
- [Using the HP NNMi–HP NA Integration](#)
- [Changing the HP NNMi–HP NA Integration](#)
- [Disabling the HP NNMi–HP NA Integration](#)
- [Troubleshooting the HP NNMi–HP NA Integration](#)
- [HP NNMi–HP NA Integration Configuration Form Reference](#)

HP NNMi–HP NA Integration

The HP NNMi–HP NA integration combines the NA configuration change detection capabilities with the NNMi network monitoring capabilities, placing more information at your fingertips when problems occur.

The integration provides for automatic configuration of devices to be managed by NNMi and for updating the NNMi configuration with information for accessing managed devices.

Without exiting NNMi, you can connect to NA to view information about NA-managed devices and configuration change events. While in NA, you can perform any NA functions for which you have the necessary credentials.

The HP NNMi–HP NA integration adds menu items to the NNMi console for opening connections to NA and for viewing configuration information on devices managed by NA. These tools provide the following functionality:

- View detailed device information, including vendor, model, modules, operating system version, and recent diagnostic results.
- View device configuration changes and configuration history.
- Compare configurations (typically the most recent and last previous configurations) to see what changed, why, and who made the changes.
- View device compliance information.
- Run NA diagnostics and command scripts from NNMi nodes.
- (NNM iSPI NET) Detect connections with mismatched speed or duplex configurations.



These features are not available for network devices that are not configured in NA or for NA devices for which change detection is disabled.

Value

The HP NNMi–HP NA integration provides the following features and benefits in an environment already running both NNMi and NA:

- Alarm integration—NNMi integration communicates NA configuration change information to the NNMi console, enabling you to quickly identify whether configuration changes may have caused network problems. From within NNMi, you can quickly access NA functionality to view specific configuration changes and device information, identify who made the change, and roll back to the previous configuration to restore network operation. Because a majority of network outages are caused by device configuration errors, this feature can enhance both problem identification and response time in resolving network downtime.
- Access to NA configuration history from NNMi—In the NNMi console, a device-level menu provides access to NA features for reviewing configuration changes. For any device in the NA database, this feature displays configuration changes side-by-side so that you can easily view changes. You can also view configuration history.
- Operations efficiency—Network operations personnel can monitor and investigate information from two data sources within a single screen.

Supported Versions

The information in this chapter applies to the following product versions:

- NA version 7.50.02 with the NA 7.50.02 service pack and the latest version of the connector
- NA version 7.60.01 with the NA 7.60.01 service pack, hotfix 107743, and the latest version of the connector
- NNMi version 9.00 or higher on the Windows, Linux, or Solaris operating system only

For information about configuring the HP NNMi–HP NA integration for an earlier version of NNMi, see the documentation that is provided with NA.

NNMi and NA can be installed on the same computer or on different computers.



For NNMi and NA to run correctly on the same computer, you must install NNMi before installing NA.

The two products can be installed on different computers in either of the following configurations:

- Different operating systems. For example, the NNMi management server is a Linux system, and the NA server is a Windows system.
- The same operating system. For example, the NNMi management server is a Windows system, and the NA server is a second Windows system.

For the most recent information about supported hardware platforms and operating systems, see the support matrices for both products.

Documentation

This chapter describes how to configure NNMi to communicate with NA and how to use the integration from the NNMi console.

The *HP Network Automation NNM Integration User's Guide*, which is included on the NA product media, describes how to use the integration from the NA user interface.

Enabling the HP NNMi–HP NA Integration

To enable the HP NNMi–HP NA integration, follow these steps:



The January 2009 edition of the *HP Network Automation NNM Integration User's Guide* for NA version 7.50 describes how to configure the integration with NNMi version 8.10. For NNMi version 8.11 or higher, the instructions in this chapter supersede those in the *HP Network Automation NNM Integration User's Guide*.

- 1 On the NNMi management server, install the NNMi connector that is provided in the latest consolidated NA patch:
 - If the NA server is a different computer from the NNMi management server, run one of the following connector installers. Choose the connector that corresponds to the operating system of the NA server:
 - `na_nnm_connector_windows.exe`
 - `na_nnm_connector_solaris.bin`
 - `na_nnm_connector_linux.bin`
 - If the NA server is the same computer as the NNMi management server, run one of the following connector installers. Choose the connector that corresponds to the operating system of the NA server:
 - `na_nnm_coresidency_windows.exe`
 - `na_nnm_coresidency_solaris.bin`
 - `na_nnm_coresidency_linux.bin`

The connector installer detects the NNMi management server and installs the components for communicating with NA on the NNMi management server. The connector installer also imports the NNMi topology into the NA database.



When the connector installer asks for the NNMi HTTP port, enter the port for connecting to the NNMi console. This port is set during NNMi installation and is specified in the following file:

- *Windows:* %NnmDataDir%\conf\nnm\props\nms-local.properties
- *UNIX:* \$NnmDataDir/conf/nnm/props/nms-local.properties

Determine the port value as follows:

- For non-SSL connections, use the value of `jboss.http.port`, which is 80 or 8004 by default (depending on the presence of another web server when NNMi was installed).
- For SSL connections, use the value of `jboss.https.port`, which is 443 by default.

2 On the NNMi management server, configure the connection between NNMi and NA:

- a In the NNMi console, open the **HP NNMi–HP NA Integration Configuration** form (**Integration Module Configuration > HP NA**).
- b Select the **Enable Integration** check box to make the remaining fields on the form available.
- c Enter the information for connecting to the NNMi management server. For information about these fields, see [NNMi Management Server Connection](#) on page 437.
- d Enter the information for connecting to the NA server. For information about these fields, see [NA Server Connection](#) on page 438.



The integration requires an HTTP connection to the NA web services. Leave the **HP NA SSL Enabled** check box cleared.

- e (NNM iSPI NET) Enter a value in the **HP NA Connection Check Interval** field. For information about this field, see [Integration Behavior](#) on page 438.
- f Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with connecting to the NA server, re-open the **HP NNMi–HP NA Integration Configuration** form (or press **ALT+LEFT ARROW** in the message window), and then adjust the values for connecting to the NA server as suggested by the text of the error message.

3 (NNM iSPI NET) If you want the integration to detect connections with mismatched speed or duplex configurations, populate the MAC addresses for the interfaces of the NNMi devices in the NA topology. Follow these steps on the NA server:

- a For each node in the NNMi topology, verify that the `NNMUuid` property is set on the corresponding device in the NA inventory.

The NNMi connector topology import process sets the `NNMUuid` property. This property is listed in the **Device Details** section of the device page in NA.

- b On the **Device Password Rule** page (**Devices > Device Tools > Device Password Rules**), create a password rule for each set of passwords for communicating with the devices in the NA inventory.
 - c On the **New Task - Discover Driver** page (**Devices > Device Tasks > Discover Driver**), discover the drivers for the devices imported from the NNMi topology.
 - d Create a snapshot of the devices imported from the NNMi topology (**Devices > Device Tasks > Take Snapshot**).
 - e Run the **NA Topology Data Gathering** diagnostic (**Devices > Device Tasks > Run Diagnostics**) for the devices imported from the NNMi topology.
 - f Verify that each device imported from the NNMi topology has a MAC address for each of its interfaces.
On a device page, click **View > Device Detail > MAC Addresses** to display the MAC addresses for that device.
- 4 *Optional.* On the NA server, configure the connection between NA and NNMi:
- a In NA, open the **Administrative Settings - 3rd Party Integrations** page (**Admin > Administrative Settings > 3rd Party Integrations**).
The **NNM Host**, **NNM HTTP Port**, **NNM Username**, and **NNM Password** fields are pre-filled with the information supplied during installation of the NNMi connector.
 - b Configure out of service behavior by selecting which device tasks generate out of service events and selecting the configuration and compliance check failure responses.
For more information, see [Disabling Network Management During Device Configuration](#) on page 432.
 - c Select the SNMP community string propagation behavior.
For more information, see [Propagating Device Community String Changes](#) on page 433.
 - d Click **Save** at the bottom of the page.

Using the HP NNMi–HP NA Integration

The HP NNMi–HP NA integration adds functionality to both NNMi and NA.

NNMi Functionality Provided by the Integration

The HP NNMi–HP NA integration provides links to NA from the NNMi console. The integration does not provide single sign-on between the products. You must enter your NA user credentials to view the NA windows.

Enabling the HP NNMi–HP NA integration adds the following actions to the NNMi console:

- **Show HP NA Diagnostic Results**—Displays a list of the NA tasks that have been scheduled for the device in an NNMi incident. Select a task to view the task results. For more information, see [Viewing the Results of Incident Actions that Access NA](#) on page 431.

- **Rerun HP NA Diagnostics**—Runs any NA actions that are configured for the device in an NNMi incident. For more information, see [Viewing the Results of Incident Actions that Access NA](#) on page 431.
- (NNM iSPI NET) **Show mismatched connections**—Displays a table of all layer 2 connections with possible speed or duplex configuration differences. For more information, see [Identifying Layer 2 Connections with Mismatched States \(NNM iSPI NET\)](#) on page 431.
- **View HP NA Device Information**—Opens the current NA **Device Details** page for the device selected in NNMi.
- **View HP NA Device Configuration**—Opens the NA **Current Configuration** page for the device selected in NNMi.



If real-time change detection is disabled for a device, the information shown is the configuration captured by NA at the last device polling interval. If configuration changes were made following that interval, the information on the **Current Configuration** page might not be the actual current configuration.

- **View HP NA Device Configuration Diffs**—Opens the NA **Compare Device Configuration** page for the device selected in NNMi.
- **View HP NA Device Configuration History**—Opens the NA **Device Configurations History** page for the device selected in NNMi.
- **View HP NA Policy Compliance Report**—Opens the NA **Policy, Rule and Compliance Search Results** page for the device selected in NNMi.
- **Telnet to HP NA Device**—Opens a **Telnet** window for connecting to the device selected in NNMi.
- **SSH to HP NA Device**—Opens an **SSH** window for connecting to the device selected in NNMi.
- **Launch HP NA**—Opens the NA user interface.
- **Launch HP NA Command Scripts**—Opens the **New Task—Run Command Script** page in NA. The page is pre-filled for the node or incident selected in the NNMi console.
- **Launch HP NA Diagnostics**—Opens the **New Task—Run Diagnostics** page in NA. The page is pre-filled for the node or incident selected in the NNMi console.

For information about using the NA functionality, see the *HP Network Automation User's Guide*.

Configuring NA Diagnostics and Command Scripts as Incident Actions

Enabling the HP NNMi–HP NA integration modifies some out-of-the-box NNMi incidents to include incident actions that access NA diagnostics each time the associated incident type occurs. [Table 40](#) lists the modified incidents.

Table 40 NNMi Incidents Configured with NA Diagnostics

NNMi Incident	NA Diagnostic
OSPFNbrStateChange	Show Neighbor
OSPFVirtIfStateChange	Show Neighbor

Table 40 NNMi Incidents Configured with NA Diagnostics (cont'd)

NNMi Incident	NA Diagnostic
OSPFIfStateChange	Show Neighbor Show Interfaces
InterfaceDown	Show Interfaces
CiscoChassisChangeNotification	Show Module

You can add an action that accesses NA to any other NNMi incident, and you can modify the default incident actions. On the **Actions** tab for an incident, add a new lifecycle transition action with **Command Type** of `ScriptOrExecutable`. In the **Command** box, enter either `naruncmdscript.ovpl` or `narundiagnostic.ovpl` with the appropriate arguments. For examples, see the action configurations of the incidents listed in [Table 40](#).

Viewing the Results of Incident Actions that Access NA

When an incident of a type that has been configured with an NA action arrives, NNMi initiates the configured action and stores the task ID of the diagnostic or command script as an attribute of that incident. The presence of the task ID enables the **Show HP NA Diagnostic Results** and **Rerun HP NA Diagnostics** items on the **Actions** menu.

To view the outcome of the action at the time the incident occurred, select the incident in any incident view, and then select **Actions > Show HP NA Diagnostic Results**.

To view current results of the configured action, select the incident in any incident view, and then select **Actions > Rerun HP NA Diagnostics**.

If you run the task multiple times, NNMi lists the most recent task ID on the **Custom Attributes** tab of the **Incident** form. The **Show HP NA Diagnostic Results** action displays all of the tasks that have been run for the incident so that you can compare the results from different runs.

Identifying Layer 2 Connections with Mismatched States (NNM iSPI NET)

If an NNM iSPI NET license key is installed on the NNMi management server and the HP NNMi—HP NA integration is enabled, NNMi periodically queries NA for the speed and duplex settings of the two interfaces on either end of each layer 2 connection in the NNMi topology. Additionally, NNMi queries NA for the speed and duplex settings of the interfaces for any new connection that is added to the NNMi topology and, when the NNM iSPI for Metrics is running, for any connection with performance threshold exceptions that might indicate a mismatched connection. NNMi uses a mismatch detection algorithm to determine whether the values might result in a mismatched connection.



NNMi is able to perform the mismatch analysis only when the NA inventory includes the MAC addresses for both interfaces that form a layer 2 connection. If the NA interface records do not include valid MAC addresses, run the **NA Topology Data Gathering** diagnostic to update the MAC address fields.

The **Actions > Show mismatched connections** command displays a table of layer 2 connections that NNMI suspects might contain speed and/or duplex mismatches, as shown in [Figure 20](#). For each suspect connection, the table lists the speed and duplex values for the interfaces on either side of the connection and NNMI's interpretation of the data. The possible interpretations are as follows:

- **MATCH** indicates that the speed values and duplex values most likely result in a properly functioning layer 2 connection.
- **POSSIBLE_MISMATCH** indicates that the speed values, the duplex values, or both speed and duplex values might potentially conflict and result in a poor or non-performing connection.
- **MISMATCH** indicates that the speed values, the duplex values, or both speed and duplex values most likely conflict and result in a poor or non-performing connection.

The **HP NA Connection Check Interval** parameter on the **HP NNMI-HP NA Integration Configuration** form specifies the frequency of the connection queries.

Figure 20 Example Mismatched Connections Table

Layer 2 Connection	Speed Comparison (configured/negotiated : configured/negotiated)	Duplex Comparison (configured/negotiated : configured/negotiated)
Small_Subnets-mlsp04[Vl137],mlsp07[Fa0/1]	MATCH (100/100 : auto-negotiated/100)	POSSIBLE_MISMATCH (full/full : auto-negotiated/half)

NA Functionality Provided by the Integration

The HP NNMI-HP NA integration provides communication from NA to NNMI (through NNMI web services calls) for the following functionality:

- [Disabling Network Management During Device Configuration](#)
- [Propagating Device Community String Changes](#)

Disabling Network Management During Device Configuration

For certain device configuration tasks, NA triggers NNMI to set the device to the **DISABLED** status during the configuration process. This administrative status suppresses NNMI monitoring of the device to prevent unnecessary incidents. Before configuring a device, NA sends an out of service event to NNMI. After device configuration succeeds, NA sends an in service event to NNMI, which removes the **DISABLED** status from the device and resumes regular state polling.

The **Out of Service Events** field on the **NA Administrative Settings - 3rd Party Integrations** page specifies the device configuration tasks that trigger NNMI to set a device to the **DISABLED** status during the task. You can select any or all of the following tasks:

- Run Command Script
- Run Diagnostics
- Update Device Software
- Deploy Passwords
- Delete ACLs

- Configure Syslog
- Reboot Device
- Discover Driver
- Run ICMP Test
- Take Snapshot
- Synchronize Startup and Running
- OS Analysis



To disable this feature, do not select any tasks.

If device configuration does not complete satisfactorily, the behavior depends on the integration configuration.

- The **If the device task fails** setting on the **NA Administrative Settings - 3rd Party Integrations** page specifies whether the integration should remove or retain the **DISABLED** status in NNMi if device configuration is not successful.
- The **If device compliance check fails after the task completed** setting on the **NA Administrative Settings - 3rd Party Integrations** page specifies whether the integration should remove or retain the **DISABLED** status in NNMi if device configuration is not compliant.

These settings apply to all device tasks selected in the **Out of Service Events** field. You cannot set the recovery behavior per task.

Propagating Device Community String Changes

When SNMP community string propagation is enabled, the integration behaves as follows:

- If the SNMP community string that NA uses for accessing a device changes, NA informs NNMi of the change. NNMi then updates the communication settings that for that device.
 NNMi immediately starts using the new community string for the device.
 NA sends updates to NNMi only when the community string for managing a device changes. NNMi does not receive updates when NA deploys a new community string to a device.
- If a new device is added to the NA inventory, NA informs NNMi of the community strings that NA uses for managing the device.

The **Propagate SNMP Community Strings** setting on the **NA Administrative Settings - 3rd Party Integrations** page specifies whether the integration should forward SNMP community strings from NA to NNMi. By default, community string propagation does not occur.

Importing NNMi 8.10 Devices into the NA Inventory

The NA inventory does not automatically update as the NNMi topology changes. After adding new devices to the NNMi topology, import those devices into the NA inventory.




Run the import command periodically to keep the NA inventory current.

To import NNMi device information into the NA inventory, follow these steps:

- 1 On the NNMi management server, change to the NA root directory. The default location is:
 - *Windows*: C:\NA
 - *UNIX*: /opt/NA
- 2 Run the following command:
 - *Windows*: nnmimport.bat
 - *UNIX*: nnmimport.sh


Changing the HP NNMi–HP NA Integration

- 1 In the NNMi console, open the **HP NNMi–HP NA Integration Configuration** form (**Integration Module Configuration > HP NA**).
- 2 Modify the values as appropriate. For information about the fields on this form, see [HP NNMi–HP NA Integration Configuration Form Reference](#) on page 437.
- 3 Verify that the **Enable Integration** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.


 The changes take effect immediately. You do not need to restart `ovjboss`.

Disabling the HP NNMi–HP NA Integration

- 1 In the NNMi console, open the **HP NNMi–HP NA Integration Configuration** form (**Integration Module Configuration > HP NA**).
- 2 Clear the **Enable Integration** check box at the top of the form, and then click **Submit** at the bottom of the form. The integration actions are no longer available.

 The changes take effect immediately. You do not need to restart `ovjboss`.

Troubleshooting the HP NNMi–HP NA Integration

 If the integration has worked successfully in the past, it is possible that some aspect of the configuration, for example, the NNMi or NA user password, has changed recently. Try updating the integration configuration as described in [HP NNMi–HP NA Integration Configuration Form Reference](#) on page 437, before walking through this entire procedure.

- 1 If the integration has never worked successfully, verify the presence of the following file:

- *Windows:* C:\NA\nnmimport.csv
- *UNIX:* /opt/NA/nnmimport.csv

If the `nnmimport.csv` file does not exist, uninstall and then reinstall the NNMi connector as described in [step 1](#) on page 427.

- 2 In the NNMi console, open the **HP NNMi–HP NA Integration Configuration** form (**Integration Module Configuration > HP NA**).

For information about the fields on this form, see [HP NNMi–HP NA Integration Configuration Form Reference](#) on page 437.

- 3 To check the status of the integration, in the **HP NNMi–HP NA Integration Configuration** form, click **Submit** at the bottom of the form (without making any configuration changes).

A new window displays a status message.

If the message indicates a problem with connecting to the NA server, NNMi and NA are not able to communicate. Continue with [step 4](#) of this procedure.

- 4 To verify the accuracy and access level of the NA credentials, log on to the NA user interface with the credentials for the **HP NA User** from the **HP NNMi–HP NA Integration Configuration** form.

If you cannot log on to the NA user interface, contact the NA administrator to verify your login credentials.

- 5 To verify that the connection to the NA server is configured correctly, in a web browser on the NNMi management server, enter the following URL:

`http://<naserver>:<naport>/soap`

Where the variables are related to values on the **HP NNMi–HP NA Integration Configuration** form as follows:

- The **HP NA SSL Enabled** check box must be cleared to indicate that connections to the NA server use the `http` protocol.
- `<naserver>` is the value of **HP NA Host**.
- `<naport>` is the value of **HP NA Port**.

If the NA web service is running on the specified server and port, the NA server responds with a message similar to:

NAS SOAP API: Only handles HTTP POST requests

- If the expected message appears, continue with [step 6](#).
- If you see an error message, the connection to the NA server is not configured correctly. Contact the NA administrator to verify the information that you are using to connect to the NA web services. Continue to troubleshoot the connection to NA until you see the expected message.

- 6 Verify that the connection to NNMi is configured correctly:

If you used the information described in this step to connect to the NNMi console in [step 2](#) of this procedure, you do not need to reconnect to the NNMi console. Continue with [step 7](#).



- a In a web browser, enter the following URL:

<protocol>://<NNMIserver>:<port>/nmm/

Where the variables are related to values on the **HP NNMi–HP NA Integration Configuration** form as follows:

- If the **HP NNMi SSL Enabled** check box is selected, <protocol> is https.
- If the **HP NNMi SSL Enabled** check box is cleared, <protocol> is http.
- <NNMIserver> is the value of **HP NNMi Host**.
- <port> is the value of **HP NNMi Port**.

- b When prompted, enter the credentials for an NNMi user with the Administrator role.

You should see the NNMi console. If the NNMi console does not appear, contact the NNMi administrator to verify the information that you are using to connect to NNMi. Continue to troubleshoot the connection to NNMi until the NNMi console appears.



You cannot sign in to the NNMi console as a user with the Web Service Client role.

- c Verify the values of the **HP NNMi User** and **HP NNMi Password**.

- If the **HP NNMi User** listed on the **HP NNMi–HP NA Integration Configuration** form has the Administrator role and you were able to connect to the NNMi console with this user name, then re-enter the corresponding password on the **HP NNMi–HP NA Integration Configuration** form.
- If the **HP NNMi User** listed on the **HP NNMi–HP NA Integration Configuration** form has the Web Service Client role, contact the NNMi administrator to verify the values of **HP NNMi User** and **HP NNMi Password**.

Passwords are hidden in the NNMi console. If you are not sure what password to specify for an NNMi user name, ask the NNMi administrator to reset the password.

- 7 Update the **HP NNMi–HP NA Integration Configuration** form with the values that you used for successful connections in [step 5](#) and [step 6](#) of this procedure.

For more information, see [HP NNMi–HP NA Integration Configuration Form Reference](#) on page 437.

- 8 Click **Submit** at the bottom of the form.
- 9 If the status message still indicates a problem with connecting to the NA server, do the following:
 - a Clear the web browser cache.
 - b Clear all saved form or password data from the web browser.
 - c Close the web browser window completely, and then re-open it.
 - d Repeat [step 7](#) and [step 8](#) of this procedure.
- 10 Test the configuration by launching one of the actions listed in [Using the HP NNMi–HP NA Integration](#) on page 429.

HP NNMi–HP NA Integration Configuration Form Reference

The **HP NNMi–HP NA Integration Configuration** form contains the parameters for configuring communications between NNMi and NA. This form is available from the **Integration Module Configuration** workspace.



Only NNMi users with the Administrator role can access the **HP NNMi–HP NA Integration Configuration** form.

The **HP NNMi–HP NA Integration Configuration** form collects information for the following general areas:

- [NNMi Management Server Connection](#)
- [NA Server Connection](#)
- [Integration Behavior](#)

To apply changes to the integration configuration, update the values on the **HP NNMi–HP NA Integration Configuration** form, and then click **Submit**.

NNMi Management Server Connection

[Table 41](#) lists the parameters for connecting to the NNMi management server. This is the same information that you use to open the NNMi console. You can determine many of these values by examining the URL that invokes an NNMi console session. Coordinate with the NNMi administrator to determine the appropriate values for this section of the configuration form.

Table 41 NNMi Management Server Information

Field	Description
HP NNMi SSL Enabled	The connection protocol specification. <ul style="list-style-type: none">• If the NNMi console is configured to use HTTPS, select the HP NNMi SSL Enabled check box. This is the default configuration.• If the NNMi console is configured to use HTTP, clear the HP NNMi SSL Enabled check box.
HP NNMi Host	The fully-qualified domain name of the NNMi management server. This field is pre-filled with host name that was used to access the NNMi console. Verify that this value is the name that is returned by the <code>nnmofficialfqdn.ovpl -t</code> command run on the NNMi management server.
HP NNMi Port	The port for connecting to the NNMi console. This field is pre-filled with the port that the jboss application server uses for communicating with the NNMi console, as specified in the following file: <ul style="list-style-type: none">• <i>Windows:</i> %NnmDataDir%\conf\nnm\props\nms-local.properties• <i>UNIX:</i> \$NnmDataDir/conf/nnm/props/nms-local.properties For non-SSL connections, use the value of <code>jboss.http.port</code> , which is 80 or 8004 by default (depending on the presence of another web server when NNMi was installed). For SSL connections, use the value of <code>jboss.https.port</code> , which is 443 by default.

Table 41 NNMi Management Server Information (cont'd)

Field	Description
HP NNMi User	The user name for connecting to the NNMi console. This user must have the NNMi Administrator or Web Service Client role.
HP NNMi Password	The password for the specified NNMi user.

NA Server Connection

Table 42 lists the parameters for connecting to the web services on the NA server. Coordinate with the NA administrator to determine the appropriate values for this section of the configuration form.

Table 42 NA Server Information

HP NA Server Parameter	Description
HP NA SSL Enabled	The connection protocol specification for connecting to the NA web services. The integration requires an HTTP connection to the NA web services. Leave the HP NA SSL Enabled check box cleared.
HP NA Host	The fully-qualified domain name or the IP address of the NA server.
HP NA Port	The port for connecting to the NA web services. The default NA ports are as follows: <ul style="list-style-type: none"> • 80—for connections to NA on a separate computer from NNMi • 8080—for connections to NA on the same computer as NNMi
HP NA User	A valid NA user account name with the NA Administrator role.
HP NA Password	The password for the specified NA user.

Integration Behavior

The **HP NA Connection Check Interval** parameter specifies the frequency with which NNMi verifies with NA the interface data for all layer 2 connections in the NNMi topology as described in [Identifying Layer 2 Connections with Mismatched States \(NNM iSPI NET\)](#) on page 431. The default interval for the connection check is 24 hours.



The connection check functionality requires an NNM iSPI NET license. If an NNM iSPI NET license key is not installed on the NNMi management server, the **HP NA Connection Check Interval** field is disabled.

HP ProCurve Manager Plus

HP ProCurve Manager Plus (PCM Plus) is a network management platform for mapping, configuring, and monitoring HP ProCurve devices. PCM Plus provides the following functionality:

- Unified management of wired and wireless Ethernet across the entire network
- Configuration, updating, monitoring, and troubleshooting of HP ProCurve devices
- Policy-based and multi-device management
- Proactive alerts with automatic alert responses
- Advanced traffic-monitoring capabilities

For information about purchasing PCM Plus, contact your HP sales representative.

This chapter contains the following topics:

- [HP NNMi–HP ProCurve Manager Plus Integration](#)
- [Using the HP NNMi–HP ProCurve Manager Plus Integration](#)

HP NNMi–HP ProCurve Manager Plus Integration

By including HP Network Node Manager i-suite Software in the PCM Plus environment, network administrators who use PCM Plus to monitor and manage their ProCurve devices gain additional insight into those devices.

The HP NNMi–HP ProCurve Manager Plus integration provides the following functionality:

- Synchronization of IPv4 ProCurve device information from the NNMi database with PCM Plus. (At this time, PCM Plus does not support IPv6 ProCurve devices.)
- Synchronization of SNMPv2 community strings and SNMPv3 user-based security model (USM) settings for communicating with the managed ProCurve devices between the two applications.

Value

The HP NNMi–HP ProCurve Manager Plus integration provides the following benefits:

- Reduces network traffic to the ProCurve devices as NNMi discovers the ProCurve devices for both NNMi and PCM Plus.
 - NNMi forwards richer ProCurve device information to PCM Plus.
 - Simplified PCM Plus network maps contain only known ProCurve devices.
- Consolidates ProCurve device event processing in NNMi.
- Reduces the cost of delivering maximum ProCurve device availability.

Supported Versions

The information in this chapter applies to the following product versions:

- PCM Plus version 3.10 with the hotfix for defect 53323
- NNMi version 9.00

NNMi and PCM Plus must be installed on separate computers. The NNMi management server and the PCM Plus computer can be of the same or different operating systems.

The PCM Plus remote agent cannot be installed on the NNMi management server.



One installation of PCM Plus version 3.10 can integrate with NNMi version 8.1x or with NNM version 7.x, but not with both products at the same time.

Documentation

The HP NNMi–HP ProCurve Manager Plus integration is fully described in Appendix A of the *HP ProCurve Manager Network Administrator's Guide*, which is available at www.procurve.com/pcm-manuals.

Using the HP NNMi–HP ProCurve Manager Plus Integration

The steps to enable the HP NNMi–HP ProCurve Manager Plus integration take place on the PCM Plus server (configuration) and the NNMi management server (installation of PCM Plus trap definitions).

Enabling the HP NNMi–HP ProCurve Manager Plus integration adds the `ICMEVT_PCMPPLUS_EVENTS_ALL` incident configuration to NNMi for all application events that PCM Plus forwards to NNMi. This incident has SNMP object ID `.1.3.6.1.4.1.11.2.3.7.11.0.63000000`.

For information about enabling, using, disabling, and troubleshooting the HP NNMi–HP ProCurve Manager Plus integration, see Appendix A of the *HP ProCurve Manager Network Administrator's Guide*.

HP RAMS MPLS WAN

The HP RAMS MPLS WAN integration enables HP Route Analytics Management System (RAMS) to support enterprises that have multiple sites connected by a WAN through ISPs that use multi-protocol label switching (MPLS) within their own networks.

For information about purchasing HP RAMS, contact your HP sales representative.

This chapter contains the following topics:

- [HP NNMi–HP RAMS MPLS WAN Integration](#)
- [Using the HP NNMi–HP RAMS MPLS WAN Integration](#)

HP NNMi–HP RAMS MPLS WAN Integration

The HP NNMi–HP RAMS MPLS WAN integration provides features for accessing MPLS WAN information from the NNMi console.

Value

The HP NNMi–HP RAMS MPLS WAN integration adds the feature to view the connectivity through different network clouds, so that NNMi users can detect and view the multiple sites connected by a WAN.

Supported Versions

The information in this chapter applies to the following product versions:

- HP RAMS version 9.00 or higher
- NNMi version 9.00 or higher, with an NNMi Advanced license

For information about the NNMi supported hardware platforms and operating systems, see the *NNMi System and Device Support Matrix*.

Documentation

The HP NNMi–HP RAMS MPLS WAN integration is fully described in *Using Route Analytics Management Systems (RAMS) with NNMi Advanced* in the NNMi help.

Using the HP NNMi–HP RAMS MPLS WAN Integration

The steps to enable the HP NNMi–HP RAMS MPLS WAN integration take place on the NNMi management server.

For information about enabling, using, disabling, and troubleshooting the HP NNMi–HP RAMS MPLS WAN integration, see *Using Route Analytics Management Systems (RAMS) with NNMi Advanced* in the NNMi help.

HP Systems Insight Manager

HP Systems Insight Manager (SIM) provides systems management of HP server and storage devices. SIM features include system discovery and identification, a single-event view, inventory data collection, and reporting.

SIM is beneficial for the following tasks:

- Troubleshooting complex problems that span server and storage infrastructure.
- Maintaining server and storage asset information.
- Modeling the impact of infrastructure and application changes before they occur.
- Tracking actual planned and unplanned changes through discovered change history.
- Gaining a shared, authoritative view of the environment through awareness of existing data repositories.
- Training network management personnel across domains of expertise.
- Shifting network management focus from daily maintenance toward future business needs.

For information about purchasing SIM, contact your HP sales representative.

This chapter contains the following topics:

- [HP NNMi–HP SIM Integration](#)
- [Enabling the HP NNMi–HP SIM Integration](#)
- [Using the HP NNMi–HP SIM Integration](#)
- [Changing the HP NNMi–HP SIM Integration Configuration](#)
- [Disabling the HP NNMi–HP SIM Integration](#)
- [Troubleshooting the HP NNMi–HP SIM Integration](#)
- [HP NNMi–HP SIM Integration Configuration Form Reference](#)

HP NNMi–HP SIM Integration

The HP NNMi–HP SIM integration provides actions for accessing several SIM tools from the NNMi console.

Value

The HP NNMi–HP SIM integration adds network device information to NNMi, so that NNMi users can detect and investigate potential network problems for HP ProLiant servers and storage devices.

Supported Versions

The information in this chapter applies to the following product versions:

- SIM version 5.30 or higher
- NNMi version 8.13 or higher

NNMi and SIM must be installed on separate computers. The NNMi management server and the SIM server computer can be of the same or different operating systems.

For the most recent information about supported hardware platforms and operating systems for NNMi, see the NNMi support matrix.

For the most recent information about supported hardware platforms and operating systems for SIM, see the quickspecs, which are available at:

www.hp.com/go/sim

Documentation

This chapter describes how to configure NNMi to communicate with SIM and how to use the integration from the NNMi console.

The SIM documentation suite describes the SIM features and capabilities in detail. The documentation suite is available for download from the SIM information library, which is available at:

www.hp.com/go/sim

Enabling the HP NNMi–HP SIM Integration

On the NNMi management server, configure the connection between NNMi and SIM by following these steps:

- 1 In the NNMi console, open the **HP NNMi–HP SIM Integration Configuration** form (**Integration Module Configuration > HP SIM**).
- 2 Select the **Enable Integration** check box to make the remaining fields on the form available.

- 3 Enter the information for connecting to the NNMi management server. For information about these fields, see [NNMi Management Server Connection](#) on page 448.
- 4 Enter the information for connecting to the SIM server. For information about these fields, see [SIM Server Connection](#) on page 448.
- 5 Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with connecting to the NNMi management server, click **Return**, and then adjust the values as suggested by the text of the error message.

- 6 Load the incident definitions for SIM-managed devices:
 - a Change to the following directory:
 - *Windows*: %NnmInstallDir%\newconfig\HPOvNmsEvent
 - *UNIX*: \$NnmInstallDir/newconfig/HPOvNmsEvent
 - b Import the SIM incident definitions by entering the following command:


```
nnmconfigimport.ovpl -f nnm-sim-incidentConfig.xml \
-u <username> -p <password>
```
- 7 Optional and recommended. Load the MIB definition files for the traps that SIM-managed devices generate:
 - a Change to the following directory:
 - *Windows*:


```
%NNM_SNMP_MIBS%\Vendor\Hewlett-Packard\SystemInsightManager
```
 - *UNIX*:


```
$NNM_SNMP_MIBS/Vendor/Hewlett-Packard/SystemInsightManager
```
 - b Use the `nnmloadmib.ovpl` command to load the appropriate MIB files for the managed environment. For example:


```
nnmloadmib.ovpl -load cpqhost.mib -u <username> -p <password>
```

 - For the HP ProLiant device traps, load the `cpqhost.mib` file, and then load the remaining `cpq*.mib` files in the `SystemInsightManager` directory.
 - For the HP Virtual Connect device traps, load the `vc*.mib` files and the `fa-mib40.mib` file into NNMi.
 - c Verify that the MIBs loaded correctly, by entering the following command:


```
nnmloadmib.ovpl -list -u <username> -p <password>
```

Using the HP NNMi–HP SIM Integration


The HP NNMi–HP SIM integration provides links from the NNMi console to the SIM agent on a device or directly to SIM. The integration does not provide single sign-on between the products. You must enter your SIM user credentials to view the SIM pages.

Enabling the HP NNMi–HP SIM integration adds the following actions to the NNMi console:

- **HP System Management Homepage**—Opens the HP System Management device home page for the node selected in the NNMi console.
- **HP Systems Insight Manager Home**—Opens the SIM home page.
- **HP Systems Insight Manager**—Opens the SIM System page for the node selected in the NNMi console.


Changing the HP NNMi–HP SIM Integration Configuration

- 1 In the NNMi console, open the **HP NNMi–HP SIM Integration Configuration** form (**Integration Module Configuration > HP SIM**).
- 2 Modify the values as appropriate. For information about the fields on this form, see [HP NNMi–HP SIM Integration Configuration Form Reference](#) on page 447.
- 3 Verify that the **Enable Integration** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.

 The changes take effect immediately. You do not need to restart `ovjboss`.

Disabling the HP NNMi–HP SIM Integration

- 1 In the NNMi console, open the **HP NNMi–HP SIM Integration Configuration** form (**Integration Module Configuration > HP SIM**).
- 2 Clear the **Enable Integration** check box at the top of the form, and then click **Submit** at the bottom of the form. The integration actions are no longer available.

 The changes take effect immediately. You do not need to restart `ovjboss`.

Troubleshooting the HP NNMi–HP SIM Integration

SIM Actions Do Not Work

If you have verified the values in the **HP NNMi–HP SIM Integration Configuration** form and you are still not able to open a SIM page from the NNMi console, do the following:

- 1 Clear the web browser cache.
- 2 Clear all saved form or password data from the web browser.
- 3 Close the web browser window completely, and then re-open it.
- 4 Re-enter the values in the **HP NNMi–HP SIM Integration Configuration** form.



Because NNMi cannot silently verify the connection to the SIM server, the **HP NNMi–HP SIM Integration Configuration** form status message applies to the NNMi management server connection information only.

- 5 Verify that SIM is running by opening the SIM homepage in a web browser.

OID Not Found in the MIB Cache Message in Traps

If the MIB definition files for the traps that SIM-managed devices generate are not loaded in NNMi, you might see an error similar to the following text:

```
<Cia .1.3.6.1.4.1.11.5.7.5.2.1.1.1.7.0 with value 1 was not found within the mib cache>
```

To resolve these errors, load the MIBs as described in [step 7](#) on page 445.

HP NNMi–HP SIM Integration Configuration Form Reference

The **HP NNMi–HP SIM Integration Configuration** form contains the parameters for configuring communications between NNMi and SIM. This form is available from the **Integration Module Configuration** workspace.



Only NNMi users with the Administrator role can access the **HP NNMi–HP SIM Integration Configuration** form.

The **HP NNMi–HP SIM Integration Configuration** form collects information for the following general areas:

- [NNMi Management Server Connection](#)
- [SIM Server Connection](#)

To apply changes to the integration configuration, update the values on the **HP NNMi–HP SIM Integration Configuration** form, and then click **Submit**.

NNMi Management Server Connection

Table 43 lists the parameters for connecting to the NNMi management server. This is the same information that you use to open the NNMi console. You can determine many of these values by examining the URL that invokes an NNMi console session. Coordinate with the NNMi administrator to determine the appropriate values for this section of the configuration form.

Table 43 NNMi Management Server Information

Field	Description
NNMi SSL Enabled	The connection protocol specification. <ul style="list-style-type: none">• If the NNMi console is configured to use HTTPS, select the NNMi SSL Enabled check box. This is the default configuration.• If the NNMi console is configured to use HTTP, clear the NNMi SSL Enabled check box.
NNMi Host	The fully-qualified domain name of the NNMi management server. This field is pre-filled with host name that was used to access the NNMi console. Verify that this value is the name that is returned by the <code>nnmofficialfqdn.ovpl -t</code> command run on the NNMi management server.
NNMi Port	The port for connecting to the NNMi console. This field is pre-filled with the port that the jboss application server uses for communicating with the NNMi console, as specified in the following file: <ul style="list-style-type: none">• <i>Windows</i>: %NnmDataDir%\conf\nnm\props\nms-local.properties• <i>UNIX</i>: \$NnmDataDir/conf/nnm/props/nms-local.properties For non-SSL connections, use the value of <code>jboss.http.port</code> , which is 80 or 8004 by default (depending on the presence of another web server when NNMi was installed). For SSL connections, use the value of <code>jboss.https.port</code> , which is 443 by default.
NNMi User	The user name for connecting to the NNMi console. This user must have the NNMi Administrator or Web Service Client role.
NNMi Password	The password for the specified NNMi user.

SIM Server Connection

Table 44 lists the parameters for connecting to the SIM server to open SIM pages. Coordinate with the SIM administrator to determine the appropriate values for this section of the configuration.

Table 44 SIM Server Information

SIM Server Parameter	Description
SIM SSL Enabled	The connection protocol specification for connecting to SIM. <ul style="list-style-type: none">• If SIM is configured to use HTTPS, select the HP SIM SSL Enabled check box. This is the default configuration.• If SIM is configured to use HTTP, clear the HP SIM SSL Enabled check box.

Table 44 SIM Server Information (cont'd)

SIM Server Parameter	Description
SIM Host	The fully-qualified domain name of the SIM server.
SIM Port	The port for connecting to SIM. If you are using the default SIM configuration, use port 50000 (for SSL connections to SIM).

nGenius Performance Manager



NetScout Systems nGenius Performance Manager provides visibility into complex networks for the following purposes:

- Application recognition and monitoring
- Analysis and troubleshooting of packets and flows
- Response time analysis
- Reporting and capacity planning
- Convergence management
- Alarming and event identification

nGenius Performance Manager leverages deep packet inspection and flow-based technologies to deliver visibility into real-time, operational intelligence that spans several types of data:

- High-level key performance indicators (KPIs), such as response time, errors, or jitter
- Application flow data, such as utilization, conversations, or top talkers
- Packet-level analysis, such as decodes and bounce diagrams

nGenius Performance Manager collects performance data from a wide variety of network data sources, allowing it to monitor the usage patterns of all network infrastructures, topologies, and applications. nGenius Performance Manager then presents the results in a collection of real-time and historical views and reports that can show the following information:

- Application performance
- Users and abusers of network resources
- Resource consumption of network capacity

For information about purchasing nGenius Performance Manager, contact your HP sales representative.

This chapter contains the following topics:

- [HP NNMi–nGenius Performance Manager Integration](#)
- [Enabling the HP NNMi–nGenius Performance Manager Integration](#)
- [Using the HP NNMi–nGenius Performance Manager Integration](#)
- [Disabling the HP NNMi–nGenius Performance Manager Integration](#)
- [Troubleshooting the HP NNMi–nGenius Performance Manager Integration](#)

HP NNMi–nGenius Performance Manager Integration

By including nGenius Performance Manager in the HP Network Node Manager i-suite Software environment, network administrators who use NNMi to monitor and manage their network devices gain application-level visibility through nGenius devices.

The HP NNMi–nGenius Performance Manager integration provides the following functionality:

- Receive nGenius Performance Manager server and probe alarms in the NNMi incident views.
- Investigate the cause of an incident by launching contextual views into nGenius Performance Manager.
- Display nGenius Probes with a NetScout icon in NNMi map views.
- Launch nGenius Performance Manager QuickViews from the NNMi console.
- Launch the nGenius Performance Manager application from the NNMi console.
- Launch NNMi Layer 2 and Layer 3 Neighbor views for the nGenius Performance Manager incidents that are available in the NNMi incident views.
- Launch the NNMi Path View map for nGenius Performance Manager incidents that are available in the NNMi incident views.
- Forward Clear Trap alarms to NNMi for each alarm generated by nGenius Performance Manager.

Value

The HP NNMi–nGenius Performance Manager integration provides the following benefits:

- Reduces the cost of delivering maximum network availability.
- Consolidates the network management infrastructure in a single console.
- Increases staff productivity and efficiency with integrated fault and application-aware performance data.
- Shrinks MTTR with contextual drill down to flow and packet-level details to identify the performance problems.

Supported Versions

The information in this chapter applies to the following product versions:

- nGenius Performance Manager version 4.6 MR1 (build 654) or higher
- nGenius K2 version shipped with nGenius Performance Manager
- CDM Agent Firmware version 4.6 MR1 (build 627) or higher
- nGenius InfiniStream appliance version 4.6 MR1 (build 631) or higher
- NNMi version 9.00 or higher

NNMi and nGenius Performance Manager Server must be installed on separate computers. The NNMi management server and the nGenius Performance Manager Server computer can be of the same or different operating systems.

Documentation

The HP NNMi–nGenius Performance Manager integration is fully described in the following document:

Integrating nGenius Performance Manager v4.3 with HP Network Node Manager i Software (NetScout part number 733-0089 Rev. A, available from <http://www.netscout.com>)

Enabling the HP NNMi–nGenius Performance Manager Integration

The nGenius Performance Manager Integration utility for HP Network Node Manager installation file is available in the following location on the nGenius Performance Manager server:

- *Windows:* %nGenius Install%\rtm\bin\nGeniusNNM8.zip
- *UNIX:* \$nGenius Install/rtm/bin/nGeniusNNM8.zip

As a user with administrative or root privileges, install the integration utility on the NNMi management server. The utility imports all NNMi integration-related configuration data for the nGenius Server into NNMi.

The high-level steps for installing the nGenius Performance Manager Integration utility are as follows. For detailed information, see *Integrating nGenius Performance Manager v4.3 with HP Network Node Manager i Software*.

- 1 Extract the installation files and configure NNMi support within nGenius Performance Manager.
- 2 Configure NetScout incidents (alarms) in NNMi.
- 3 Configure the nGenius Probe to send SNMP traps to port 395.
- 4 *Optional.* Configure Probe Router Mapping.

Using the HP NNMi–nGenius Performance Manager Integration

For information about using the HP NNMi–nGenius Performance Manager integration, see *Integrating nGenius Performance Manager v4.3 with HP Network Node Manager i Software*.

Disabling the HP NNMi–nGenius Performance Manager Integration

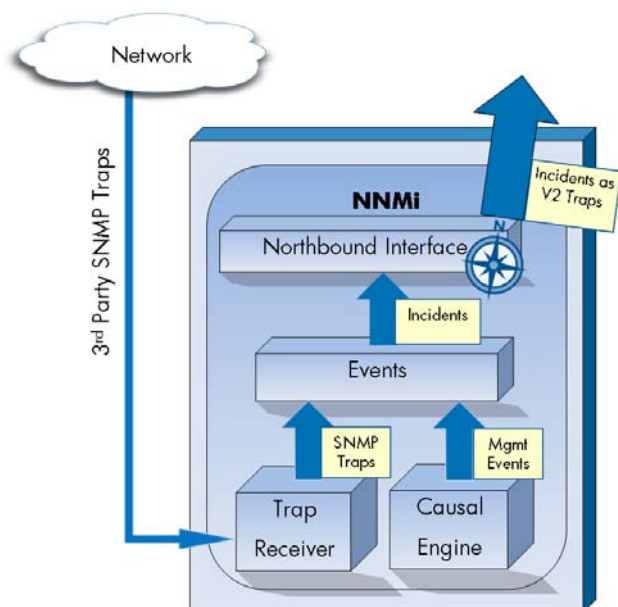
For information about disabling the HP NNMi–nGenius Performance Manager integration, see *Integrating nGenius Performance Manager v4.3 with HP Network Node Manager i Software*.

Troubleshooting the HP NNMi–nGenius Performance Manager Integration

For information about troubleshooting the HP NNMi–nGenius Performance Manager integration, contact NetScout Systems Customer Support. Contact information is available at:

<http://www.netscout.com/support>

NNMi Northbound Interface



NNMi provides the NNMi northbound interface for forwarding NNMi incidents to any application that can receive SNMPv2c traps. For each NNMi management server, you can implement the NNMi northbound interface to multiple northbound applications, each configured separately.

NNMi includes support for using the NNMi northbound interface to integrate with the following products:

- The Operations Management functionality of the HP Business Service Management (BSM) platform; for information, see [HP BSM Operations Management](#) on page 471.
- The HP Operations Manager (HPOM) active messages browser; for information, see [HP NNMi—HPOM Integration \(Agent Implementation\)](#) on page 487.
- IBM Tivoli Netcool/OMNIbus; for information, see [HP NNMi Integration Module for Netcool Software](#) on page 525.

To integrate with a different northbound application, follow the instructions in this chapter.

This chapter contains the following topics:

- [NNMi Northbound Interface](#)
- [Enabling the NNMi Northbound Interface](#)
- [Using the NNMi Northbound Interface](#)
- [Changing the NNMi Northbound Interface](#)
- [Disabling the NNMi Northbound Interface](#)
- [Troubleshooting the NNMi Northbound Interface](#)
- [Application Failover and the NNMi Northbound Interface](#)
- [NNMi Northbound Interface Destination Form Reference](#)

NNMi Northbound Interface

The NNMi northbound interface forwards NNMi management events as SNMPv2c traps to a northbound application. The northbound application might filter, act on, and display the NNMi traps. The northbound application might also provide tools for accessing the NNMi console in the context of an NNMi trap.

The NNMi northbound interface can send incident lifecycle state change notifications, incident correlation notifications, and incident deletion notifications to the northbound application. In this way, the northbound application can replicate the results of NNMi causal analysis.

The NNMi northbound interface can also forward the SNMP traps that NNMi receives to the northbound application. The NNMi northbound interface does not forward events generated by NNM 6.x or 7.x management stations to the northbound application.

Value

The NNMi northbound interface enables event consolidation in a third-party or custom event consolidator. The NNMi northbound interface enriches events with information that can be used to integrate other applications with NNMi.

Supported Versions

The information in this chapter applies to NNMi version 9.00 or higher.

For the most recent information about supported hardware platforms and operating systems, see the NNMi support matrix.

Terminology

This chapter uses the following terms:

- Northbound application—Any application that can receive and process SNMPv2c traps.
- Trap-receiving component—The portion of a northbound application that receives SNMP traps.
 - Some applications include a separately installable component that receives SNMP traps and forwards them to another component for processing.
 - For any northbound application that does not include such a component, “trap-receiving component” is synonymous with “northbound application.”
- NNMi northbound interface—The NNMi functionality that forwards NNMi incidents as SNMPv2c traps to a northbound application.
- Northbound destination—One configuration of the NNMi northbound interface that defines the connection to the trap-receiving component of a northbound application and specifies the types of traps that NNMi will send to that northbound application.

This chapter describes how to configure NNMi to forward NNMi incidents to any northbound application. For information about a particular northbound application, see that application's documentation.

Enabling the NNMi Northbound Interface



NNMi does not limit the amount of information sent in an SNMP trap using UDP. If any network hardware in the transmission path cannot handle the size of the trap data, or if network traffic is heavy, the trap might be lost. Therefore, it is recommended that the trap-receiving component of the northbound application be installed on the NNMi management server. The northbound application is responsible for ensuring reliable information transfer.

NNMi Northbound Interface Upgraded from NNMi 8.1x

Upgrading to NNMi 9.00 moves the NNMi northbound interface configuration to the **HP NNMi—Northbound Interface Destination** form. (To access this form, click **Integration Module Configuration > Northbound Interface**, select a destination, and then click **Edit**.) Verify that the upgraded configuration is accurate for the current northbound application.



If the northbound destination identifies an HP Operations Manager (HPOM) agent, the following suggestion applies:

Prior to NNMi version 9.00, the agent implementation of the HP NNMi—HPOM integration was enabled on the **HP NNMi—Northbound Interface Configuration** form. For long-term maintenance purposes, it is recommended that you reconfigure the integration on the **HP NNMi—HPOM Agent Destination** form. After this reconfiguration, delete the destination from the **HP NNMi—Northbound Interface Destination** form. For more information, see [HP NNMi—HPOM Integration \(Agent Implementation\)](#) on page 487.

[Table 45](#) describes the mapping of the NNMi northbound interface configuration parameters from NNMi 8.13 to NNMi 9.00. For information about these fields, see [NNMi Northbound Interface Destination Form Reference](#) on page 466.

Table 45 NNMi Northbound Interface Configuration Parameters from NNMi 8.13 to NNMi 9.00

NNMi 8.13 Northbound Interface Parameter	NNMi 9.00 Northbound Interface Parameter
Use NNMi FQDN	Not selected maps to the Use Loopback option or the Other option of the Host field. Selected maps to the NNMi FQDN option of the Host field.
Trap Destination Host	Maps to the Host field.
Trap Destination Port	Maps to the Port field.
Trap Community String	Maps to the Community String field.

Table 45 NNMi Northbound Interface Configuration Parameters from NNMi 8.13 to NNMi 9.00

NNMi 8.13 Northbound Interface Parameter	NNMi 9.00 Northbound Interface Parameter
Holding period (minutes)	No direct mapping. The Dampening tab on the incident configuration form defines the holding time per incident, or use <code>nnmsetdampenedinterval.ovpl</code> to set the same holding time for all incident configurations. For more information, see the <code>nnmsetdampenedinterval.ovpl</code> reference page, or the UNIX manpage.
Send 3rd Party Traps	Not selected maps to Management incidents. Selected maps to Both incidents.
NNMi uses https	Maps to the NNMi Console Access field.
Allow Loopback	If a non-loopback interface exists for the hostname or IP address, maps to the NNMi FQDN option or the Other option of the Host field. If the configured hostname or IP address has a loopback-only interface, maps to the Use Loopback option of the Host field.
Trap Destination IP Address	Same.
Uptime (seconds)	Same.
Last Sweep Time	Not available. No change to integration behavior.
NNMi URL	Same.
Dampened Stream Enabled All Stream Enabled	Fields not available. No change to integration behavior.
None	Other fields are set to the default behavior for the NNMi northbound interface as described in NNMi Northbound Interface Integration Content on page 467.

New NNMi Northbound Interface Configuration

To enable the NNMi northbound interface, follow these steps:

- 1 If necessary, configure the northbound application to understand the NNMi trap definitions.
- 2 On the NNMi management server, configure NNMi incident forwarding:
 - a In the NNMi console, open the **HP NNMi–Northbound Interface Destinations** form (**Integration Module Configuration > Northbound Interface**), and then click **New**.
(If you have selected an available destination, click **Reset** to make the **New** button available.)
 - b Select the **Enabled** check box to make the remaining fields on the form available.
 - c Enter the information for connecting to the northbound application.
For information about these fields, see [Northbound Application Connection Parameters](#) on page 466.

- d Specify the sending options and incident filter for which content to send to the northbound application.

For information about these fields, see [NNMi Northbound Interface Integration Content](#) on page 467.

- e Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with the settings, click **Return**, and then adjust the values as suggested by the text of the error message.

- 3 *Optional.* Create contextual interaction with NNMi by creating URLs that provide access to NNMi views from the northbound application.

For information, in the NNMi console, click **Help > NNMi Documentation Library > Integrate NNMi Elsewhere with URLs**.

Using the NNMi Northbound Interface

When the NNMi northbound interface is enabled, the northbound destination determines the information that NNMi sends to a northbound application. Configure the northbound application to display and interpret the forwarded traps, as appropriate in your network environment. For complete information about the contents and format of the traps that NNMi sends to a northbound application, see the `hp-nnmi-nbi.mib` file.

NNMi sends only one copy of each management event, SNMP trap, or notification trap to a northbound destination. NNMi does not queue traps. If the trap-receiving component of a northbound application is unavailable when NNMi forwards a trap, that trap is lost.

This section describes the types of traps the integration can send. For information about setting the content configuration, see [NNMi Northbound Interface Integration Content](#) on page 467.

Incident Forwarding

Management events

When the northbound destination includes management events, NNMi forwards each management event incident to the northbound application when that incident changes to the REGISTERED lifecycle state.

The OID of the forwarded management event is the SNMP Object ID on the **Management Event Configuration** form in the NNMi console. NNMi forwards all custom management events with the OID 1.3.6.1.4.1.11.2.17.19.2.0.9999.

Third-party SNMP traps

When the northbound destination includes third-party SNMP traps, NNMi forwards each incoming SNMPv1, v2c, or v3 format trap to the northbound application when the associated incident changes to the REGISTERED lifecycle state. NNMi preserves the original trap varbinds in order (as defined in the MIB) and appends the NNMi-specific varbinds to the message payload. If the original trap does not contain all of the defined varbinds, NNMi pads NULL values for the missing varbinds. If the MIB is not loaded in NNMi, NNMi cannot correctly reconstruct the trap and append the NNMi incident data; therefore, NNMi does not forward this trap.

For third-party SNMP traps, note the following:

- Because NNMi reconstructs a trap from its SNMP trap incident, the forwarded trap is in SNMPv2c format regardless of the format in which NNMi received the original trap.
- The forwarded SNMP trap shows the NNMi management server as the trap source. To determine the original trap source, examine the values of the (n+21)th varbind, IncidentNodeHostname (1.3.6.1.4.1.11.2.17.19.2.2.21) and the (n+24)th varbind, IncidentNodeMgmtAddr (1.3.6.1.4.1.11.2.17.19.2.2.24), where n is the number of varbinds defined for the trap in the MIB.

If any of the devices that NNMi manages also send traps to the northbound application, the northbound application must manage the duplicate device traps.

For a comparison of trap forwarding mechanisms, see [Trap and Incident Forwarding](#) on page 87.

Incident Lifecycle State Change Notifications

Enhanced closed traps

When the northbound destination includes enhanced closed notifications, NNMi sends an EventLifecycleStateClosed (1.3.6.1.4.1.11.2.17.19.2.0.1000) trap to the northbound application when the lifecycle state of an incident changes to CLOSED in NNMi. The EventLifecycleStateClosed trap includes much of the data from the original incident. The previous lifecycle state value is not included. The EventLifecycleStateClosed trap identifies the original incident in the sixth varbind, IncidentUuid (1.3.6.1.4.1.11.2.17.19.2.2.6).

State change traps

When the northbound destination includes lifecycle state changed notifications, NNMi sends a LifecycleStateChangeEvent (1.3.6.1.4.1.11.2.17.19.2.0.1001) trap to the northbound application when the lifecycle state of an incident changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state in NNMi. The northbound application can associate the LifecycleStateChangeEvent with the original incident.

The LifecycleStateChangeEvent trap identifies the original incident and the lifecycle state change in the following varbinds:

- IncidentUuid, the sixth varbind (1.3.6.1.4.1.11.2.17.19.2.2.6)
This value matches the value of the sixth varbind in a management event or the (n+6)th varbind in a third-party SNMP trap varbind.
- IncidentLifecycleStatePreviousValue, the seventh varbind (1.3.6.1.4.1.11.2.17.19.2.2.200)
- IncidentLifecycleStateCurrentValue, the eighth varbind (1.3.6.1.4.1.11.2.17.19.2.2.201)

The following table lists the possible integer values for lifecycle state.

Name	Integer Value
registered	1
inprogress	2

Name	Integer Value
completed	3
closed	4
dampened	5

Incident Correlation Notifications

When the northbound destination includes incident correlation notifications, NNMi sends incident correlation traps to the northbound application as NNMi causal analysis correlates incidents. The northbound application can use the information in the traps to replicate the correlation changes.

Single correlation traps

For the single correlation trap option, the integration sends the following correlation traps:

- EventDedupCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1100)
- EventImpactCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1101)
- EventPairwiseCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1102)
- EventRateCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1103)
- EventApaCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1104)
- EventCustomCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1105)

Each trap identifies one parent-child incident correlation relationship in the following varbinds:

- IncidentCorrelationIndicatorParentUuid, the sixth varbind (1.3.6.1.4.1.11.2.17.19.2.2.6)
- IncidentCorrelationIndicatorChildUuid, the seventh varbind (1.3.6.1.4.1.11.2.17.19.2.2.300)

Group correlation traps

For the group correlation option, the integration sends the following correlation traps:

- EventDedupCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2100)
- EventImpactCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2101)
- EventPairwiseCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2102)
- EventRateCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2103)
- EventApaCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2104)
- EventCustomCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2105)

Each trap identifies the parent-child incident correlation relationships in the following varbinds:

- IncidentCorrelationIndicatorParentUuid, the sixth varbind (1.3.6.1.4.1.11.2.17.19.2.2.6)
- IncidentCorrelationIndicatorChildCount, the seventh varbind (1.3.6.1.4.1.11.2.17.19.2.2.301)
- IncidentCorrelationIndicatorChildUuidCsv, the eighth varbind (1.3.6.1.4.1.11.2.17.19.2.2.302)

This value is a comma-separated-value list of child incident UUIDs.

Incident Deletion Notifications

When the northbound destination includes incident deletion notifications, NNMi sends an EventDeleted (1.3.6.1.4.1.11.2.17.19.2.0.3000) trap to the northbound application when an incident is deleted in NNMi. The EventDeleted trap identifies the original incident in the sixth varbind, IncidentUuid (1.3.6.1.4.1.11.2.17.19.2.2.6).

Event Forwarding Filter

When the northbound destination includes an incident filter, the object identifiers (OIDs) in the filter include or exclude (depending on the selected configuration option) the following event types:

- NNMi management event incidents
- Third-party SNMP traps
- EventLifecycleStateClosed traps
- LifecycleStateChangeEvent traps
- EventDeleted traps
- Correlation notification traps

The following notes apply to correlation notification traps:

- If the incident filter prevents the forwarding of the parent incident for a correlation, NNMi does not send a correlation notification trap to the northbound application.
- If the incident filter prevents the forwarding of a child incident for a correlation, the forwarded correlation notification trap does not include that child incident's UUID. (If the correlation notification trap would not contain any child incident UUIDs, NNMi does not send that trap to the northbound application.)
- The DuplicateCorrelation management event is forwarded independently of the EventDedupCorrelation or EventDedupCorrelationGroup correlation notification traps. Likewise, the RateCorrelation management event is forwarded independently of the EventRateCorrelation or EventRateCorrelationGroup correlation notification traps. If the incident filter prevents the forwarding of one of these correlation notification traps, NNMi might still forward the associated management events.

Changing the NNMi Northbound Interface

To change the NNMi northbound interface configuration parameters, follow these steps:

- 1 In the NNMi console, open the **HP NNMi–Northbound Interface Destinations** form (**Integration Module Configuration > Northbound**).
- 2 Select a destination, and then click **Edit**.
- 3 Modify the values as appropriate.
For information about the fields on this form, see [NNMi Northbound Interface Destination Form Reference](#) on page 466.
- 4 Verify that the **Enabled** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

Disabling the NNMi Northbound Interface

No SNMP trap queuing occurs while a northbound destination is disabled.

To discontinue the forwarding of NNMi incidents to a northbound application, follow these steps:



- 1 In the NNMi console, open the **HP NNMi–Northbound Interface Destinations** form (**Integration Module Configuration > Northbound**).
- 2 Select a destination, and then click **Edit**.
Alternatively, click **Delete** to entirely remove the configuration for the selected destination.
- 3 Clear the **Enabled** check box at the top of the form, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

Troubleshooting the NNMi Northbound Interface

If the NNMi northbound interface is not working as expected, follow these steps until you have resolved the problem:

- 1 Verify that the trap destination port is not blocked by a firewall.
Ensure that the NNMi management server is able to directly address the northbound application by host and port.
- 2 Verify that the integration is running correctly:
 - a In the NNMi console, open the **HP NNMi–Northbound Interface Destinations** form (**Integration Module Configuration > Northbound**).
 - b Select a destination, and then click **Edit**.

- c Verify that the **Enabled** check box is selected.
- 3 If the northbound destination includes management events, verify this functionality:
- a In the **Closed Key Incidents** view of the NNMi console, open any incident.
 - b Set the incident lifecycle state to **Registered**, and then click  **Save**.
 - c Set the incident lifecycle state to **Closed**, and then click  **Save and Close**.
 - d After 30 seconds, determine whether the northbound application received an EventLifecycleStateClosed trap (or a LifecycleStateChangeEvent trap) for this incident should.
 - If the northbound application received the trap, continue with [step 4](#).
 - If the northbound application did not receive the trap, configure a new northbound destination to connect with a different northbound application, and then repeat this test from [step a](#).

If the repeated test succeeds, the problem is with the first northbound application. Consult that application's documentation for troubleshooting information.

If the repeated test fails, contact HP Support for assistance.

- 4 If the northbound destination includes SNMP traps, verify this functionality:
- a Generate an SNMP trap against a node in the NNMi topology by entering the following command on the NNMi management server:

```
nnmsnmpnotify.ovpl -u username -p password -a \  
discovered_node NNMi_node linkDown
```

Where *discovered_node* is the hostname or IP address of a node in the NNMi topology and *NNMi_node* is the hostname or IP address of the NNMi management server.

- b After 30 seconds, determine whether the northbound application received the forwarded trap.
 - If the northbound application received the trap, the NNMi northbound interface is working correctly.
 - If the northbound application did not receive the trap, configure a new northbound destination to connect with a different northbound application, and then repeat this test from [step a](#).

If the repeated test succeeds, the problem is with the first northbound application. Consult that application's documentation for troubleshooting information.

If the repeated test fails, contact HP Support for assistance.

Application Failover and the NNMi Northbound Interface

If the NNMi management server will participate in NNMi application failover, the information in this topic applies to any integration that implements the NNMi northbound interface for sending traps to a northbound application.

The traps that NNMi sends to a northbound application include the NNMi URL in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2). Traps received prior to application failover reference what is now the standby NNMi management server. When the URL points to the standby NNMi management server, any actions that use the URL value (for example, launching the NNMi console) will fail.

Local Northbound Application

If the trap-receiving component of the northbound application is located on the NNMi management server, the following considerations apply to the configuration of the NNMi northbound interface:

- The trap-receiving component of the northbound application must be installed and configured identically on the active and standby NNMi management servers. Configure SNMP trap reception on the same port on both NNMi management servers.
- Configure the NNMi northbound interface on the primary NNMi management server only.

On the **HP NNMi–Northbound Interface Destination** form, select either the **NNMi FQDN** or the **Use Loopback** option for **Host** identification.

At startup, the NNMi northbound interface determines the correct name or IP address of the current NNMi management server. In this way, the northbound interface sends traps to the trap-receiving component of the northbound application on the active NNMi management server.

Remote Northbound Application

If the trap-receiving component of the northbound application is not located on the NNMi management server, configure the NNMi northbound interface on the primary NNMi management server only. On the **HP NNMi–Northbound Interface Destination** form, select the **Other** option for **Host** identification.

NNMi Northbound Interface Destination Form Reference

The **HP NNMi–Northbound Interface Destination** form contains the parameters for configuring communications between NNMi and a northbound application. This form is available from the **Integration Module Configuration** workspace. (On the **HP NNMi–Northbound Interface Destinations** form, click **New**; or select a destination, and then click **Edit**.)



Only NNMi users with the Administrator role can access the **HP NNMi–Northbound Interface Destination** form.

The **HP NNMi–Northbound Interface Destination** form contains information for the following areas:

- [Northbound Application Connection Parameters](#) on page 466
- [NNMi Northbound Interface Integration Content](#) on page 467
- [NNMi Northbound Interface Destination Status Information](#) on page 469

To apply changes to the integration configuration, update the values on the **HP NNMi–Northbound Interface Destination** form, and then click **Submit**.

Northbound Application Connection Parameters

[Table 46](#) lists the parameters for configuring the connection to the northbound application.

Table 46 Northbound Application Connection Information

Field	Description
Host	<p>The fully-qualified domain name (preferred) or the IP address of the server on which the trap-receiving component of the northbound application runs.</p> <p>The integration supports the following methods for identifying the server:</p> <ul style="list-style-type: none">• NNMi FQDN NNMi manages the connection to the northbound application on the NNMi management server and the Host field becomes read-only. This is the recommended configuration for northbound applications on the NNMi management server.• Use Loopback NNMi manages the connection to the northbound application on the NNMi management server and the Host field becomes read-only.• Other Enter a hostname or IP address for identifying the northbound application server in the Host field. NNMi validates that the hostname or IP address in the Host field is not configured as a loopback adapter. This is the default configuration. <p>NOTE: If the NNMi management server participates in NNMi application failover, see Application Failover and the NNMi Northbound Interface on page 465 for information about the impact of application failover on the integration.</p>

Table 46 Northbound Application Connection Information (cont'd)

Field	Description
Port	<p>The UDP port where the northbound application receives SNMP traps. Enter the port number specific to the northbound application.</p> <p>NOTE: If the trap-receiving component of the northbound application is on the NNMi management server, this port number must be different from the port on which NNMi receives SNMP traps, as set in the SNMP Port field on the Communication Configuration form in the NNMi console.</p>
Community String	<p>A read-only community string for the northbound application to receive traps. If the northbound application configuration requires a community string in the received SNMP traps, enter that value.</p> <p>If the northbound application configuration does not require a specific community string, use the default value, which is <code>public</code>.</p>

NNMi Northbound Interface Integration Content

Table 47 lists the parameters for configuring which content the NNMi northbound interface sends to the northbound application.

Table 47 NNMi Northbound Interface Content Configuration Information

Field	Description
Incidents	<p>The incident forwarding specification.</p> <ul style="list-style-type: none"> • Management NNMi forwards only NNMi-generated management events to the northbound application. • 3rd Party SNMP Trap NNMi forwards only SNMP traps that NNMi receives from managed devices to the northbound application. • Both NNMi forwards to the northbound application both NNMi-generated management events and SNMP traps that NNMi receives from managed devices. This is the default configuration. <p>NNMi begins forwarding incidents as soon as you enable the northbound destination.</p> <p>For more information, see Incident Forwarding on page 459.</p>

Table 47 NNMi Northbound Interface Content Configuration Information (cont'd)

Field	Description
Lifecycle State Changes	<p>The incident change notification specification.</p> <ul style="list-style-type: none">• Enhanced Closed NNMi sends an incident closed trap to the northbound application for each incident that changes to the CLOSED lifecycle state. This is the default configuration.• State Changed NNMi sends an incident lifecycle state changed trap to the northbound application for each incident that changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state.• Both NNMi sends an incident closed trap to the northbound application for each incident that changes to the CLOSED lifecycle state. Additionally, the integration sends an incident lifecycle state changed trap to the northbound application for each incident that changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state. NOTE: In this case, each time an incident changes to the CLOSED lifecycle state, the integration sends two notification traps: an incident closed trap and an incident lifecycle state changed trap. <p>For more information, see Incident Lifecycle State Change Notifications on page 460.</p>
Correlations	<p>The incident correlation notification specification.</p> <ul style="list-style-type: none">• None NNMi does not notify the northbound application of incident correlations resulting from NNMi causal analysis. This is the default configuration.• Single NNMi sends a trap for each parent-child incident correlation relationship resulting from NNMi causal analysis.• Group NNMi sends one trap per correlation that lists all child incidents correlated to a parent incident. <p>For more information, see Incident Correlation Notifications on page 461.</p>
Deletions	<p>The incident deletion specification.</p> <ul style="list-style-type: none">• Don't Send NNMi does not notify the northbound application when incidents are deleted in NNMi. This is the default configuration.• Send NNMi sends a deletion trap to the northbound application for each incident that is deleted in NNMi. <p>For more information, see Incident Deletion Notifications on page 462.</p>

Table 47 NNMi Northbound Interface Content Configuration Information (cont'd)

Field	Description
NNMi Console Access	<p>The connection protocol specification in the URL for browsing to the NNMi console from the northbound application. The traps that NNMi sends to the northbound application include the NNMi URL in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2).</p> <p>The configuration page defaults to the setting that matches the NNMi configuration. If the NNMi console is configured to accept both HTTP and HTTPS connections, you can change the HTTP connection protocol specification in the NNMi URL. For example, if all users of the northbound application are on the intranet, you can set NNMi console access from the northbound application to be over HTTP. To change the protocol for connecting to the NNMi console from the northbound application, select the HTTP option or the HTTPS option as appropriate.</p>
Incident Filters	<p>A list of object identifiers (OIDs) on which the integration filters the events sent to the northbound application. Each filter entry can be a valid numeric OID (for example, .1.3.6.1.6.3.1.1.5.4.1.3.6.1.4.1.9) or OID prefix (for example, .1.3.6.1.6.3.1.1.5.*).</p> <p>Select one of the following options:</p> <ul style="list-style-type: none"> • None NNMi sends all events to the northbound application. This is the default configuration. • Include NNMi sends only the specific events that match the OIDs identified in the filter. • Exclude NNMi sends all events except for the specific events that match the OIDs identified in the filter. <p>Specify the incident filter:</p> <ul style="list-style-type: none"> • To add a filter entry, enter the text in the lower text box, and then click Add. • To delete a filter entry, select that entry from the list in the upper box, and then click Remove. <p>For more information, see Event Forwarding Filter on page 462.</p>

NNMi Northbound Interface Destination Status Information

Table 48 lists the read-only status information for the northbound destination. This information is useful for verifying that the integration is working correctly.

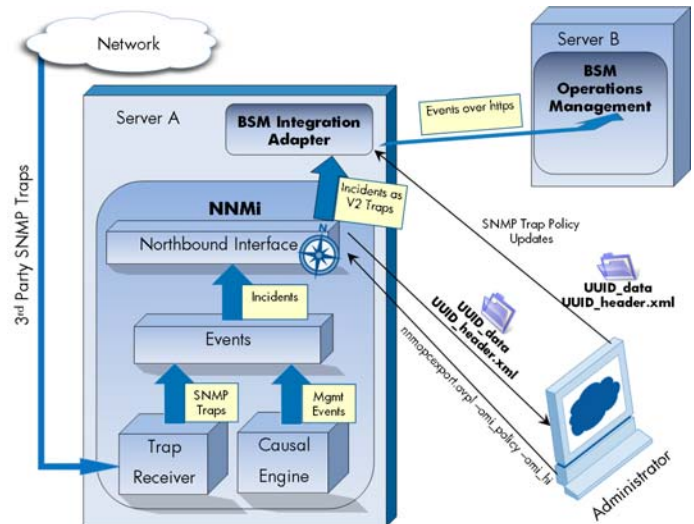
Table 48 NNMi Northbound Interface Destination Status Information

Field	Description
Trap Destination IP Address	<p>The IP address to which the destination host name resolves. This value is unique to this northbound destination.</p>

Table 48 NNMi Northbound Interface Destination Status Information (cont'd)

Field	Description
Uptime (seconds)	<p>The time (in seconds) since the northbound component was last started. The traps that NNMi sends to a northbound application include this value in the sysUptime field (1.3.6.1.2.1.1.3.0).</p> <p>This value is the same for all integrations that use the NNMi northbound interface. To see the latest value, either refresh or close and re-open the form.</p>
NNMi URL	<p>The URL for connecting to the NNMi console. The traps that NNMi sends to a northbound application include this value in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2).</p> <p>This value is unique to this northbound destination.</p>

HP BSM Operations Management



The Operations Management functionality of the HP Business Service Management (BSM) platform provides comprehensive event management; proactive performance monitoring; and automated alerting, reporting, and graphing for management operating systems, middleware, and application infrastructure. BSM Operations Management consolidates events from a wide range of sources into a single view.

For information about purchasing BSM, contact your HP sales representative.

This chapter describes the direct integration between NNMi and the BSM Operations Management event browser. If you want NNMi incidents to appear in the HPOM active messages browser *as well as* the BSM Operations Management event browser, you do not need to read this chapter. Instead, do *both* of the following in any order:

- Configure the agent implementation of the HP NNMi–HPOM integration, as described in [HP NNMi–HPOM Integration \(Agent Implementation\)](#) on page 487.
- Configure the HPOM integration with the BSM Operations Management event browser, as described in the *HP OMi Installation and Deployment Guide*.

This chapter contains the following topics:

- [HP NNMi–HP BSM Operations Management Integration](#) on page 472
- [Enabling the HP NNMi–HP BSM Operations Management Integration](#) on page 473
- [Using the HP NNMi–HP BSM Operations Management Integration](#) on page 476
- [Changing the HP NNMi–HP BSM Operations Management Integration](#) on page 479
- [Disabling the HP NNMi–HP BSM Operations Management Integration](#) on page 480
- [Troubleshooting the HP NNMi–HP BSM Operations Management Integration](#) on page 480
- [HP NNMi–HPOM Agent Destination Form Reference \(BSM Operations Management Integration\)](#) on page 483

HP NNMi—HP BSM Operations Management Integration

The HP NNMi—HP BSM Operations Management integration forwards NNMi management events as SNMPv2c traps to the HP BSM Integration Adapter on the NNMi management server. The BSM Integration Adapter filters the NNMi traps and forwards them to the BSM Operations Management event browser.

The HP NNMi—HP BSM Operations Management integration can also forward the SNMP traps that NNMi receives to the adapter. The integration does not forward events generated by NNM 6.x or 7.x management stations to the adapter.

The HP NNMi—HP BSM Operations Management integration also provides for accessing the NNMi console from within the BSM Operations Management event browser.



This chapter describes the direct integration between NNMi and the BSM Operations Management event browser. If you want NNMi incidents to appear in the HPOM active messages browser as well as the BSM Operations Management event browser, you do not need to read this chapter. Instead, do *both* of the following in any order:

- Configure the agent implementation of the HP NNMi—HPOM integration, as described in [HP NNMi—HPOM Integration \(Agent Implementation\)](#) on page 487.
- Configure the HPOM integration with the BSM Operations Management event browser as described in the *HP OMi Installation and Deployment Guide*.

The HP NNMi—HP BSM Operations Management integration is a specific implementation of the NNMi northbound interface, which is described in [NNMi Northbound Interface](#) on page 455.

The HP NNMi—HP BSM Operations Management integration consists of the following components:

- nnmi-hpom agent integration module
- nnmopcexport.ovpl tool

Value

The HP NNMi—HP BSM Operations Management integration provides event consolidation in the BSM Operations Management event browser for the network management, system management, and application management domains, so that users of the BSM Operations Management event browser can detect and investigate potential network problems.

The primary features of the integration are as follows:

- Automatic incident forwarding from NNMi to the BSM Integration Adapter. Forwarded incidents appear in the BSM Operations Management event browser.
- Access to the NNMi console from the BSM Operations Management event browser.
 - Open the NNMi **Incident** form in the context of a selected event.
 - Open an NNMi view (for example, the Layer 2 Neighbor view) in the context of a selected event and node.
 - Launch an NNMi tool (for example, status poll) in the context of a selected event and node.

Supported Versions

The information in this section applies to the following product versions:

- BSM version 9.00 with the HP Operations Manager i license
- NNMi version 9.00 on the Windows or Linux operating system only

NNMi and BSM must be installed on separate computers. The NNMi management server and the BSM server computer can be of the same or different operating systems.

The BSM Integration Adapter requires a license and must be installed on the NNMi management server computer *after* NNMi installation.

For the most recent information about supported hardware platforms and operating systems, see the support matrices for all products.

Documentation

This chapter describes how to configure NNMi to communicate with the BSM Operations Management event browser.

The BSM documentation describes how to install and use the BSM Integration Adapter and the applications that access the NNMi console from the BSM Operations Management event browser.

- *HP BSM Integration Adapter Installation and Configuration Guide*
- *HP BSM Integration Adapter User Guide*
- HP BSM Integration Adapter help
- *HP BSM Operations Management Extensibility Guide*

Enabling the HP NNMi—HP BSM Operations Management Integration

It is recommended that an experienced BSM Integration Adapter user complete the procedure for enabling the HP NNMi—HP BSM Operations Management integration.



When NNMi integrates with the HP Business Service Management (BSM) topology database, the HP NNMi—HP BSM Operations Management integration can associate incidents regarding NNMi-managed devices with BSM configuration items (CIs). This information is not available with the standard NNMi northbound interface. For more information, see [Configuration Item Identifiers](#) on page 476.

To enable the HP NNMi—HP BSM Operations Management integration, follow these steps:

- 1 On the NNMi management server, generate an SNMP trap policy file for the traps that NNMi forwards:
 - a Verify that the NNMi services are running:

```
ovstatus -c
```

All NNMi services should show the state RUNNING.

- b Generate the SNMP trap policy file by entering the following command:

```
nnmopcexport.ovpl -u <username> -p <password> \  
-template "NNMi Management Events" -application "NNMi" \  
-omi_policy -omi_hi
```

The values for `<username>` and `<password>` correspond to an NNMi console user with the Administrator role.

This command creates two files in the current directory:

- The `<UUID>_data` file is the SNMP trap policy file, where `<UUID>` is a universally unique identifier.
- The `<UUID>_header.xml` file identifies the `<UUID>_data` file to the BSM Integration Adapter.



Do not edit or rename these output files, as doing so renders them unusable by the BSM Integration Adapter.

The SNMP trap policy file includes a policy condition for each management event and SNMP trap configuration in the current NNMi incident configuration. For information about customizing the output of this command, see the *nnmopcexport.ovpl* reference page, or the UNIX manpage.

For information about the default policy conditions and customizing conditions, see [Using the HP NNMi—HP BSM Operations Management Integration](#) on page 476.

- 2 On the NNMi management server, configure the BSM Integration Adapter:

- a Install and configure the BSM Integration Adapter as described in the *HP BSM Integration Adapter Installation and Configuration Guide*.



The HP Operations agent from HPOM and the BSM Integration Adapter cannot run simultaneously on one system. If necessary, uninstall the HP Operations agent before installing the BSM Integration Adapter.

- b Use the BSM Integration Adapter user interface to import the header and policy files created in [step 1](#) of this procedure.

For more information, see *Managing policies > Import policies* in the HP BSM Integration Adapter help.

- 3 Identify an available port for SNMP communications between NNMi and the BSM Integration Adapter.

The BSM Integration Adapter will listen on this port for the SNMP traps that NNMi forwards to this port. While enabling the integration, this port number is used in both [step 4](#) (for the BSM Integration Adapter) and [step 5](#) (for NNMi) of this procedure.



The SNMP communications port is different from the HTTP port for the Apache Tomcat server that you specified while configuring the BSM Integration Adapter with the `ia-config.bat` (Windows) or `ia-config.sh` (Linux) command.

Because the BSM Integration Adapter is installed on the NNMi management server, this port number must be different from the port on which NNMi receives SNMP traps.

- a In the NNMi console, open the **Communication Configuration** form from the **Configuration** workspace.
- b In the **Default SNMP Settings** area, notice the value of **SNMP Port**.

- c Select a port that is different from the value of **SNMP Port** on the **Communication Configuration** form. A good practice is to use a port number similar to 162, which is the standard UDP port for receiving SNMP traps. For example, if port 162 is not available, try port 5162.
 - d On the NNMi management server, run the command `netstat -a`, and then search the output for the port you selected in [step c](#). If that port number does not appear in the output, it is probably available for the BSM Integration Adapter to use.
- 4 On the NNMi management server, configure the agent inside the BSM Integration Adapter with a custom port for receiving SNMP traps from NNMi by entering the following commands:
- *Windows* NNMi management server:
 - a Configure the agent:


```
ovconfchg -ns eaagt -set SNMP_TRAP_PORT <custom_port> \
          -set SNMP_SESSION_MODE NNM_LIBS
```
 - b Restart the agent:


```
ovc -restart opctrapi
```
 - *Linux* NNMi management server:
 - a Configure the agent:


```
ovconfchg -ns eaagt -set SNMP_TRAP_PORT <custom_port> \
          -set SNMP_SESSION_MODE NO_TRAPD
```
 - b Restart the agent:


```
ovc -restart opctrapi
```

For `<custom_port>`, use the port that you identified in [step 3](#) of this procedure.

- 5 On the NNMi management server, configure NNMi incident forwarding to the BSM Integration Adapter:
- a In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
 - b Click **HPOM agent implementation**, and then click **New**.
(If you have selected an available destination, click **Reset** to make the **New** button available.)
 - c On the **HP NNMi–HPOM Agent Destination** form, select the **Enabled** check box to make the remaining fields on the form available.
 - d Enter the information for connecting to the BSM Integration Adapter on the NNMi management server. The trap destination port is the port that you identified in [step 3](#) of this procedure.

For information about these fields, see [BSM Integration Adapter Connection](#) on page 483.
 - e Specify the sending options. Select the **HTTP** option for the **NNMi Console Access** field.

For information about these fields, see [BSM Operations Management Integration Content](#) on page 484.

- f Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with the settings, click **Return**, and then adjust the values as suggested by the text of the error message.

- 6 *Optional.* On the BSM server, install and configure the HPOprInf infrastructure content pack.

For information, see the *HP BSM Operations Management Extensibility Guide*.

Using the HP NNMi—HP BSM Operations Management Integration

The HP NNMi—HP BSM Operations Management integration provides a one-way flow of NNMi management events and SNMP traps to the BSM Operations Management event browser. The NNMi SNMP trap policy determines how the BSM Operations Management event browser treats and displays the incoming traps. For example, you can change a policy condition to include the value of a trap custom attribute in the event title.



NNMi sends only one copy of each management event or SNMP trap to the BSM Integration Adapter. This behavior is different from that of the NNMi 6.x/7.x integration with HPOM.

View the forwarded NNMi incidents in the BSM Operations Management event browser. Menu commands in the BSM Operations Management event browser provide access to NNMi views in the context of the selected event. Information embedded in each event supports this cross-navigation:

- The `nnmi.server.name` and `nnmi.server.port` custom attributes in the event identify the NNMi management server.
- The `nnmi.incident.uuid` custom attribute identifies the incident in the NNMi database.

In the BSM Operations Management event browser, the original trap source appears in the **Object** field on the **Additional Info** tab and in the `nnm.source.name` custom attribute.

Configuration Item Identifiers

In HP Business Service Management (BSM) and HP Universal CMDB Software (UCMDB), a configuration item (CI) is a database representation of a component in the IT environment. A CI can be a line of business, business process, application, server hardware, or a service.

When NNMi integrates with the BSM topology database or UCMDB, NNMi shares CI information with BSM or UCMDB for the devices that NNMi manages. In this case, the HP NNMi—HP BSM Operations Management integration can associate incidents regarding NNMi-managed devices with BSM or UCMDB CIs. The SNMP trap policy conditions enable this association.

For information about the integrations with BSM and UCMDB, see:

- [HP Business Service Management Topology](#) on page 415
- [HP Universal CMDB](#) on page 423

Health Indicators

Because the NNMi SNMP trap policy file was created with the `-omi_hi` option to `nnmopcexport.ovpl`, the policy file associates a health indicator with each standard NNMi management event in the SNMP trap policy file, as appropriate. (Not all management event types have health indicators.) The health indicator is available in the `EtiHint` custom attribute.

For the specific health indicators, see the SNMP trap policy file.

Default Policy Conditions

The default integration behavior varies with the integration content, as described here:

- NNMi management event incidents
 - The NNMi SNMP trap policy file includes conditions for all NNMi management event configurations defined in the NNMi incident configuration when the file was generated.
 - The events created from NNMi management events appear in the BSM Operations Management event browser.
 - These traps include the CI information described in [Configuration Item Identifiers](#) on page 476.
 - The events created from these traps include health indicators described in [Health Indicators](#) on page 477.
- Third-party SNMP traps
 - The NNMi SNMP trap policy file includes conditions for all SNMP trap configurations defined in the NNMi incident configuration when the file was generated.
 - The events created from third-party traps appear in the BSM Operations Management event browser.
 - These traps include the CI information described in [Configuration Item Identifiers](#) on page 476.
 - The events created from these traps do not include health indicators.
 - If you configure the integration to forward all received SNMP traps and the BSM Operations Management event browser receives SNMP traps directly from devices that NNMi manages, the BSM Operations Management event browser receives duplicate device traps. You can set the policies to correlate SNMP traps from NNMi with those that the BSM Operations Management event browser receives directly from managed devices.

- EventLifecycleStateClosed traps
 - The BSM Integration Adapter logs the events created from these traps. Generally, they do not appear in the BSM Operations Management event browser.
 - The NNMi SNMP trap policy file causes the BSM Integration Adapter to acknowledge the event that corresponds to the closed NNMi incident in the BSM Operations Management event browser.
- LifecycleStateChangeEvent traps
 - The NNMi SNMP trap policy file does not include conditions for processing these traps. The BSM Integration Adapter does not forward these traps to the BSM Operations Management event browser.
- EventDeleted traps
 - The NNMi SNMP trap policy file does not include conditions for processing these traps. The BSM Integration Adapter does not forward these traps to the BSM Operations Management event browser.
- Correlation notification traps
 - The BSM Integration Adapter logs the events created from these traps. They do not appear in the BSM Operations Management event browser.
 - The BSM Integration Adapter processes the NNMi correlation traps to replicate NNMi incident correlation in the BSM Operations Management event browser.

Customizing Policy Conditions

Use the BSM Integration Adapter user interface to customize the default policy conditions. For more information, see *Developing SNMP interceptor policies > Configure SNMP rules* in the HP BSM Integration Adapter help.

More Information

For more information about the HP NNMi—HP BSM Operations Management integration, see the following references:

- For descriptions of the trap types that the integration sends to the BSM Integration Adapter, see [Using the NNMi Northbound Interface](#) on page 459.
- For information about the format of the traps that NNMi sends to the BSM Integration Adapter, see the `hp-nnmi-nbi.mib` file.
- For detailed information about using the HP NNMi—HP BSM Operations Management integration, see the *HP BSM Operations Management Extensibility Guide*.

Changing the HP NNMi—HP BSM Operations Management Integration

Update the SNMP Trap Policy Conditions for New NNMi Traps

If new SNMP trap incident configurations have been added to NNMi since the integration was configured, follow these steps:

- 1 On the NNMi management server, use the `nnmopcexport.ovpl` command to create an SNMP trap policy file for the new traps.

For the `-template` option, specify a name that is different from the names of the existing SNMP trap policy files.

Use the `-omi_policy` and `-omi_hi` options.

You can limit the file contents to a specific author or OID prefix value. For more information, see the `nnmopcexport.ovpl` reference page, or the UNIX manpage.

- 2 Use the BSM Integration Adapter user interface to import the new header and policy files.

Alternatively, you can re-create the SNMP trap policy file for all NNMi management events and SNMP traps. If you take this approach, importing the new files into the BSM Integration Adapter overwrites any existing policy customizations.

Change the Configuration Parameters

To change the integration configuration parameters, follow these steps:

- 1 In the NNMi console, open the **HP NNMi—HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **HPOM agent implementation**.
- 3 Select a destination, and then click **Edit**.
- 4 Modify the values as appropriate.

For information about the fields on this form, see [HP NNMi—HPOM Agent Destination Form Reference \(BSM Operations Management Integration\)](#) on page 483.

- 5 Verify that the **Enable Integration** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

Disabling the HP NNMi—HP BSM Operations Management Integration

No SNMP trap queuing occurs while a destination is disabled.

To discontinue the forwarding of NNMi incidents to the BSM Integration Adapter, follow these steps:

- 1 In the NNMi console, open the **HP NNMi—HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **HPOM agent implementation**.
- 3 Select a destination, and then click **Edit**.
Alternatively, click **Delete** to entirely remove the configuration for the selected destination.
- 4 Clear the **Enable Integration** check box at the top of the form, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

Optionally deactivate or delete the SNMP trap policy as described in the HP BSM Integration Adapter help.

Troubleshooting the HP NNMi—HP BSM Operations Management Integration

BSM Operations Management Event Browser Contains No Forwarded Incidents



In the following procedure, the `OVBIN` environment variable refers to the `bin` directory containing the commands for configuring the agent inside the BSM Integration Adapter. The `OVBIN` environment variable defaults to the following value:

- *Windows:* `<drive>\Program Files\HP\HP BTO Software\bin`
- *Linux:* `/opt/OV/bin`

If the BSM Operations Management event browser does not contain any incidents from NNMi, follow these steps:

- 1 On the NNMi management server, verify the agent configuration:
 - *Windows* NNMi management server:
`%OVBIN%\ovconfget eaagt`
 - *Linux* NNMi management server:
`$OVBIN/ovconfget eaagt`

The command output should include the following information:

- *Windows:*

```
SNMP_SESSION_MODE=NNM_LIBS
SNMP_TRAP_PORT=<custom_port>
```

- *Linux:*

```
SNMP_SESSION_MODE=NO_TRAPD
SNMP_TRAP_PORT=<custom_port>
```

The value of *<custom_port>* should *not* be 162 and should match the value of the **Port** field on the **HP NNMI-HPOM Agent Destination** form.

- 2 Evaluate the agent configuration by considering the results from [step 1](#):
 - If the agent configuration is as expected, continue with [step 3](#) of this procedure.
 - If the `SNMP_SESSION_MODE` parameter is not set correctly, repeat [step 4](#) on page 475 until the `ovconfigget` command returns the expected results.
 - If the value of *<custom_port>* is 162 or does not match the value of the **Port** field on the **HP NNMI-HPOM Agent Destination** form, repeat [step 3](#) on page 474 through [step 5](#) on page 475, as appropriate, until the `ovconfigget` command returns the expected results.

- 3 On the NNMI management server, verify that the agent is running:

- *Windows* NNMI management server:

```
%OVBIN%\opcagt -status
```

- *Linux* NNMI management server:

```
$(OVBIN)/opcagt -status
```

The command output should include an `opctrapi` entry similar to the following example:

```
opctrapi OVO SNMP Trap Interceptor AGENT,EA (4971) Running
```

If the output is not as expected, restart the agent:

```
ovc -restart opctrapi
```

- 4 On the NNMI management server, verify that the agent is listening on the expected SNMP trap port:

- a Run the following command:

```
— Windows: netstat -an | findstr <custom_port>
```

```
— Linux: netstat -an | grep <custom_port>
```

Where *<custom_port>* is the value of `SNMP_TRAP_PORT` from [step 1](#) of this procedure.

- b Verify that the output includes the state `LISTENING` or `LISTEN`.

If the output is not as expected, restart the agent:

```
ovc -restart opctrapi
```

- 5 On the NNMi management server, verify that the SNMP trap policy file for NNMi has been deployed to the BSM Integration Adapter on the NNMi management server:

- *Windows* NNMi management server:

```
%OVBIN%\ovpolicy -list
```

- *Linux* NNMi management server:

```
$OVBIN/ovpolicy -list
```

The command output should include an entry similar to the following example:

Type	Name	Status	Version
trapi	"NNMi Management Events"	enabled	0001.0000

The value of the Name field is the name of the SNMP trap policy file from the `-template` option to `nnmopcexport.ovpl` in [step 1](#) on page 473.

- 6 Verify that the BSM Integration Adapter is receiving traps:
 - a Verify that the BSM Integration Adapter can send events to the BSM Operations Management event browser.
 - b Enable tracing of the BSM Integration Adapter to determine whether the traps arrive at the BSM Integration Adapter.
- 7 Verify that NNMi is forwarding management events to the BSM Integration Adapter.

For information, see [Troubleshooting the NNMi Northbound Interface](#) on page 463.

BSM Operations Management Event Browser Contains Only Some Forwarded Incidents

If one or more NNMi incidents do not appear in the BSM Operations Management event browser, follow these steps:

- 1 On the NNMi management server, verify that the SNMP trap policy does not suppress the trap.

- 2 On the BSM server, verify that BSM Operations Management is running.

If the BSM server shuts down, the BSM Integration Adapter queues received traps. The BSM Integration Adapter forwards the queued traps when the BSM Operations Management event browser becomes available.

If the BSM Integration Adapter shuts down, the forwarded traps are lost. NNMi does not resend traps.

- 3 On the NNMi management server, verify that the NNMi processes are running:

```
ovstatus -c
```

Any traps sent to NNMi while it is shut down are lost.

HP NNMi-HPOM Agent Destination Form Reference (BSM Operations Management Integration)

The **HP NNMi-HPOM Agent Destination** form contains the parameters for configuring communications between NNMi and the BSM Integration Adapter. This form is available from the **Integration Module Configuration** workspace. (On the **HP NNMi-HPOM Integration Selection** form, click **HPOM agent implementation**. Click **New**, or select a destination, and then click **Edit**.)



Only NNMi users with the Administrator role can access the **HP NNMi-HPOM Agent Destination** form.

The **HP NNMi-HPOM Agent Destination** form collects information for the following areas:

- [BSM Integration Adapter Connection](#) on page 483
- [BSM Operations Management Integration Content](#) on page 484
- [BSM Integration Adapter Destination Status Information](#) on page 486

To apply changes to the integration configuration, update the values on the **HP NNMi-HPOM Agent Destination** form, and then click **Submit**.

BSM Integration Adapter Connection

[Table 49](#) lists the parameters for configuring the connection to the BSM Integration Adapter.

Table 49 BSM Integration Adapter Connection Information

Field	Description
Host	<p>The fully-qualified domain name (preferred) or the IP address of the NNMi management server, which is the system on which the BSM Integration Adapter receives SNMP traps from NNMi.</p> <p>The integration supports the following methods for identifying the BSM Integration Adapter host:</p> <ul style="list-style-type: none">• NNMi FQDN NNMi manages the connection to the BSM Integration Adapter on the NNMi management server and the Host field becomes read-only. This is the default and recommended configuration.• Use Loopback Do not use this option.• Other Do not use this option. <p>NOTE: If the NNMi management server participates in NNMi application failover, see Application Failover and the NNMi Northbound Interface on page 465 for information about the impact of application failover on the integration module.</p>

Table 49 BSM Integration Adapter Connection Information (cont'd)

Field	Description
Port	<p>The UDP port where the BSM Integration Adapter receives SNMP traps. Enter the port number specific to the BSM Integration Adapter. This value is the port that you identified in step 3 on page 474.</p> <p>To determine the port, run the <code>ovconfget -eaagt</code> command on the NNMi management server. The trap port is the value of the <code>SNMP_TRAP_PORT</code> variable.</p> <p>NOTE: This port number must be different from the port on which NNMi receives SNMP traps, as set in the SNMP Port field on the Communication Configuration form in the NNMi console.</p>
Community String	<p>A read-only community string for the BSM Integration Adapter to receive traps. For the HP NNMi—HP BSM Operations Management integration, use the default value, which is <code>public</code>.</p>

BSM Operations Management Integration Content

[Table 50](#) lists the parameters for configuring which content NNMi sends to the BSM Integration Adapter.

Table 50 BSM Operations Management Integration Content Configuration Information

Field	Description
Incidents	<p>The incident forwarding specification.</p> <ul style="list-style-type: none"> • Management NNMi forwards only NNMi-generated management events to the BSM Integration Adapter. • SNMP 3rd Party Trap NNMi forwards only SNMP traps that NNMi receives from managed devices to the BSM Integration Adapter. • Both NNMi forwards to the BSM Integration Adapter both NNMi-generated management events and SNMP traps that NNMi receives from managed devices. This is the default configuration. <p>NNMi begins forwarding incidents as soon as you enable the destination. For more information, see Incident Forwarding on page 459.</p>

Table 50 BSM Operations Management Integration Content Configuration Information (cont'd)

Field	Description
Lifecycle State Changes	<p>The incident change notification specification.</p> <ul style="list-style-type: none">• Enhanced Closed NNMi sends an incident closed trap to the BSM Integration Adapter for each incident that changes to the CLOSED lifecycle state. This is the default configuration.• State Changed NNMi sends an incident lifecycle state changed trap to the BSM Integration Adapter for each incident that changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state.• Both NNMi sends an incident closed trap to the BSM Integration Adapter for each incident that changes to the CLOSED lifecycle state. Additionally, the integration sends an incident lifecycle state changed trap to the BSM Integration Adapter for each incident that changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state. NOTE: In this case, each time an incident changes to the CLOSED lifecycle state, the integration sends two notification traps: an incident closed trap and an incident lifecycle state changed trap. <p>For more information, see Incident Lifecycle State Change Notifications on page 460.</p>
Correlations	<p>The incident correlation notification specification.</p> <ul style="list-style-type: none">• None NNMi does not notify the BSM Integration Adapter of incident correlations resulting from NNMi causal analysis. This is the default configuration.• Single NNMi sends a trap for each parent-child incident correlation relationship resulting from NNMi causal analysis.• Group NNMi sends one trap per correlation that lists all child incidents correlated to a parent incident. <p>For more information, see Incident Correlation Notifications on page 461.</p>
Deletions	<p>The incident deletion specification.</p> <ul style="list-style-type: none">• Don't Send NNMi does not notify the BSM Integration Adapter when incidents are deleted in NNMi. This is the default configuration.• Send NNMi sends a deletion trap to the BSM Integration Adapter for each incident that is deleted in NNMi. <p>For more information, see Incident Deletion Notifications on page 462.</p>
NNMi Console Access	<p>The connection protocol specification in the URL for browsing to the NNMi console from the BSM Operations Management event browser. The traps that NNMi sends to the BSM Integration Adapter include the NNMi URL in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2).</p> <p>The integration requires an HTTP connection to the NNMi console. Select the HTTP option.</p>

Table 50 BSM Operations Management Integration Content Configuration Information (cont'd)

Field	Description
Incident Filters	<p>A list of object identifiers (OIDs) on which the integration filters the events sent to the BSM Integration Adapter. Each filter entry can be a valid numeric OID (for example, .1.3.6.1.6.3.1.1.5.4.1.3.6.1.4.1.9) or OID prefix (for example, .1.3.6.1.6.3.1.1.5.*).</p> <p>Select one of the following options:</p> <ul style="list-style-type: none"> • None NNMi sends all events to the BSM Integration Adapter. This is the default configuration. • Include NNMi sends only the specific events that match the OIDs identified in the filter. • Exclude NNMi sends all events except for the specific events that match the OIDs identified in the filter. <p>Specify the incident filter:</p> <ul style="list-style-type: none"> • To add a filter entry, enter the text in the lower text box, and then click Add. • To delete a filter entry, select that entry from the list in the upper box, and then click Remove. <p>For more information, see Event Forwarding Filter on page 462.</p>

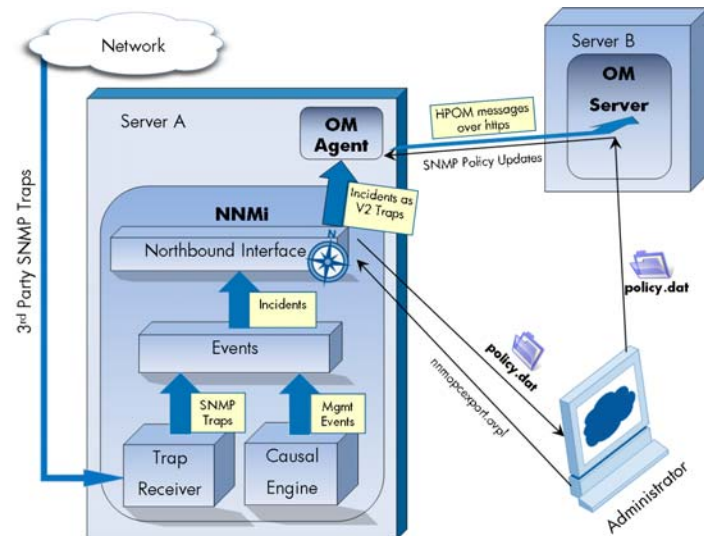
BSM Integration Adapter Destination Status Information

Table 51 lists the read-only status information for the BSM Integration Adapter. This information is useful for verifying that the integration is working correctly.

Table 51 BSM Integration Adapter Destination Status Information

Field	Description
Trap Destination IP Address	<p>The IP address to which the BSM Integration Adapter destination host name resolves.</p> <p>This value is unique to this destination.</p>
Uptime (seconds)	<p>The time (in seconds) since the northbound component was last started. The traps that NNMi sends to the BSM Integration Adapter include this value in the sysUptime field (1.3.6.1.2.1.1.3.0).</p> <p>This value is the same for all integrations that use the NNMi northbound interface. To see the latest value, either refresh or close and re-open the form.</p>
NNMi URL	<p>The URL for connecting to the NNMi console. The traps that NNMi sends to the BSM Integration Adapter include this value in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2).</p> <p>This value is unique to this northbound destination.</p>

HP Operations Manager



HP Operations Manager (HPOM) provides comprehensive event management; proactive performance monitoring; and automated alerting, reporting, and graphing for management operating systems, middleware, and application infrastructure. HPOM consolidates events from a wide range of sources into a single view.

For information about purchasing HPOM, contact your HP sales representative.

This chapter describes the available integrations:

- [HP NNMi—HPOM Integration \(Agent Implementation\)](#)
- [HP NNMi—HPOM Integration \(Web Services Implementation\)](#)

HP NNMi—HPOM Integration (Agent Implementation)

The agent implementation of the HP NNMi—HPOM integration is the preferred solution for integrating HPOM with NNMi.



This chapter describes the integration between NNMi and the HPOM active messages browser. If you want NNMi incidents to appear in the BSM version 9.00 Operations Management event browser *only*, you do not need to read this chapter. Instead, see [HP BSM Operations Management](#) on page 471.

If you want NNMi incidents to appear in the BSM version 9.00 Operations Management event browser or the HP Operations Manager i (HP OMi) version 8.10 event browser *as well as* the HPOM active messages browser, do *both* of the following in any order:

- Configure the agent implementation of the HP NNMi—HPOM integration, as described in this chapter.
- Configure the HPOM integration with the BSM Operations Management event browser or the HP OMi event browser, as described in the *HP OMi Installation and Deployment Guide*.

If the agent and the web services implementations of the HP NNMi—HPOM integration both forward messages to the same HPOM management server, you might not see all messages from both implementations in the HPOM active messages browser. For this reason, HP does not support running both implementations of the HP NNMi—HPOM integration from one NNMi management server to the same HPOM management server concurrently.

This section contains the following topics:

- [About the HP NNMi—HPOM Integration \(Agent Implementation\)](#) on page 488
- [Enabling the HP NNMi—HPOM Integration \(Agent Implementation\)](#) on page 490
- [Using the HP NNMi—HPOM Integration \(Agent Implementation\)](#) on page 494
- [Changing the HP NNMi—HPOM Integration Configuration \(Agent Implementation\)](#) on page 497
- [Disabling the HP NNMi—HPOM Integration \(Agent Implementation\)](#) on page 498
- [Troubleshooting the HP NNMi—HPOM Integration \(Agent Implementation\)](#) on page 498
- [HP NNMi—HPOM Agent Destination Form Reference \(Agent Implementation\)](#) on page 502

About the HP NNMi—HPOM Integration (Agent Implementation)

The agent implementation of the HP NNMi—HPOM integration forwards NNMi management events as SNMPv2c traps to an HP Operations agent on the NNMi management server. The agent filters the NNMi traps and forwards them to the HPOM active messages browser. The agent configuration determines which HPOM management server receives the forwarded incident.

The HP NNMi—HPOM integration can also forward the SNMP traps that NNMi receives to the agent. The integration does not forward events generated by NNMi 6.x or 7.x management stations to the agent.

The HP NNMi—HPOM integration also provides for accessing the NNMi console from within HPOM.

The agent implementation of the HP NNMi—HPOM integration is a specific implementation of the NNMi northbound interface, which is described in [NNMi Northbound Interface](#) on page 455.

The agent implementation of the HP NNMi—HPOM integration consists of the following components:

- `nnmi-hpom agent integration module`
- `nnmopcxport.ovpl tool`

Value

The HP NNMi—HPOM integration provides event consolidation in the HPOM active messages browser for the network management, system management, and application management domains, so that HPOM users can detect and investigate potential network problems.

The primary features of the integration are as follows:

- Automatic incident forwarding from NNMi to the HP Operations agent. Forwarded incidents appear in the HPOM active messages browser.
- Access to the NNMi console from HPOM.
 - HPOM users can open the NNMi **Incident** form in the context of a selected message.
 - HPOM users can launch an NNMi view (for example, the Layer 2 Neighbor view) in the context of a selected message and node.
 - HPOM users can launch an NNMi tool (for example, status poll) in the context of a selected message and node.

Supported Versions

The information in this section applies to the following product versions:

- HPOM for Windows version 8.10 or higher
- HPOM for UNIX version 8.30 or higher (including HPOM for UNIX version 9.00 or higher)
- HPOM for Linux version 9.00 or higher
- NNMi version 9.00 or higher

NNMi and HPOM must be installed on separate computers. The NNMi management server and the HPOM management server computer can be of the same or different operating systems.

The HP Operations agent requires a license and must be installed on the NNMi management server computer *after* NNMi installation.

For the most recent information about supported hardware platforms and operating systems, see the support matrices for all products.

Documentation

This chapter describes how to configure NNMi to communicate with HPOM.

The HPOM documentation describes how to install and use the HPOM applications that access the NNMi console from the HPOM active messages browser.

- For HPOM for Windows, see the information for the HP NNMi Adapter in the HPOM help.
- For HPOM for UNIX version 9.xx, see the *Integrating NNMi into HPOM* section in the *HP Operations Manager for UNIX Administrator's Reference*.
- For HPOM for UNIX version 8.3x, see the *HP NNMi-HPOM Integration for HP Operations Manager User's Guide*.
- For HPOM for Linux, see the *Integrating NNMi into HPOM* section in the *HP Operations Manager for Linux Administrator's Reference*.

Enabling the HP NNMi-HPOM Integration (Agent Implementation)

It is recommended that an experienced HPOM administrator complete the procedure for enabling the agent implementation of the HP NNMi-HPOM integration.



When NNMi integrates with the HP Business Service Management (BSM) topology database, the agent implementation of the HP NNMi-HPOM integration can associate incidents regarding NNMi-managed devices with BSM configuration items (CIs). This information is not available with the standard NNMi northbound interface. For more information, see [Configuration Item Identifiers](#) on page 494.

This section describes the following procedures:

- [Integration Configuration Upgraded from NNMi 8.1x](#) on page 490
- [New Integration Configuration](#) on page 490

Integration Configuration Upgraded from NNMi 8.1x

Prior to NNMi version 9.00, the agent implementation of the HP NNMi-HPOM integration was enabled on the **HP NNMi—Northbound Interface Configuration** form. For information about how the NNMi 8.1x configuration maps to NNMi 9.00, see [NNMi Northbound Interface Upgraded from NNMi 8.1x](#) on page 457.

For long-term maintenance purposes, it is recommended that you reconfigure the integration on the **HP NNMi-HPOM Agent Destination** form. After this reconfiguration, delete the destination from the **HP NNMi—Northbound Interface Destination** form.

New Integration Configuration

To enable agent implementation of the HP NNMi-HPOM integration, follow these steps:

- 1 On the NNMi management server, generate an SNMP trap policy file:

- a Verify that the NNMi services are running:

```
ovstatus -c
```

All NNMi services should show the state RUNNING.

- b Generate the SNMP trap policy file by entering the following command:

```
nnmopcexport.ovpl -u <username> -p <password> \  
-template "NNMi Management Events" -application "NNMi" \  
-file NNMi_policy.dat
```

The values for **<username>** and **<password>** correspond to an NNMi console user with the Administrator role.



If HPOM will forward the NNMi incidents to the HP OMi event browser or to the BSM Operations Management event browser, also use the **-omi_hi** option to add health indicators to the management event policy conditions. For more information, see [Health Indicators](#) on page 495.

The SNMP trap policy file includes a policy condition for each management event and SNMP trap configuration in the current NNMi incident configuration. For information about customizing the output of this command, see the *nnmopcexport.ovpl* reference page, or the UNIX manpage.

For information about the default policy conditions and customizing conditions, see [Using the HP NNMi–HPOM Integration \(Agent Implementation\)](#) on page 494.

- 2 On the HPOM management server, configure HPOM to receive messages from NNMi:
 - a In the HPOM console, add a node for the NNMi management server.
 - b Install the HP Operations agent on the NNMi management server.
 - c Transfer the `NNMi_policy.dat` file created in [step 1](#) of this procedure from the NNMi management server to the HPOM management server.
 - d Import the `NNMi_policy.dat` file into HPOM.
 - *HPOM for Windows*: Use the `ImportPolicies` command.
 - *HPOM for UNIX* version 9.x: Use the `opcpolicy` command.
 - *HPOM for UNIX* version 8.x: Use the `opctempl` command.
 - *HPOM for Linux*: Use the `opcpolicy` command.
 - e Deploy the `NNMi Management Events` policy to the NNMi managed node.
 - f In the HPOM console, add an external node to catch all forwarded NNMi incidents.

For initial testing, set the node filter to `<*>.<*>.<*>.<*>` (for an IP filter) or `<*>` (for a name filter). After you validate the integration, restrict the external node filter to match your network.



If you do not set up an HPOM managed node for an NNMi incident source node, the HPOM management server discards all incidents regarding that node.

For more information, see the following references:

- *HPOM for Windows*:
 - *Import OVO for UNIX templates* in the HPOM help
 - *Configuring external nodes* in the HPOM help
- *HPOM for UNIX*:
 - *HP Operations Manager for UNIX HTTPS Agent Concepts and Configuration Guide*
 - *HP Operations Manager for UNIX Concepts Guide*
 - *HP Operations Manager for UNIX Administrator's Reference*
 - *HP Operations Manager for UNIX Developer's Toolkit Developer's Reference*
 - *opcnod(1M)*, *opcbbcdist(1M)*, *opcragt(1M)*, *opcfigupl(1M)*, *opcpolicy(1M)* (version 9.xx), and *opctempl(1M)* (version 8.3x) manpages
- *HPOM for Linux*:
 - *HP Operations Manager for Linux HTTPS Agent Concepts and Configuration Guide*
 - *HP Operations Manager for Linux Concepts Guide*
 - *HP Operations Manager for Linux Administrator's Reference*

- *HP Operations Manager for Linux Developer's Toolkit Developer's Reference*
 - *opnode(1M)*, *opcbcdist(1M)*, *opcragt(1M)*, *opccfgupl(1M)*, and *opcpolicy(1M)* manpages
- 3 Identify an available port for SNMP communications between NNMi and the HP Operations agent.

The HP Operations agent will listen on this port for the SNMP traps that NNMi forwards to this port. While enabling the integration, this port number is used in both [step 4](#) (for the HP Operations agent) and [step 5](#) (for NNMi) of this procedure.

Because the HP Operations agent is installed on the NNMi management server, this port number must be different from the port on which NNMi receives SNMP traps.

- a In the NNMi console, open the **Communication Configuration** form from the **Configuration** workspace.
 - b In the **Default SNMP Settings** area, notice the value of the **SNMP Port**.
 - c Select a port that is different from the value on the **Communication Configuration** form. A good practice is to use a port number similar to 162, which is the standard UDP port for receiving SNMP traps. For example, if port 162 is not available, try port 5162.
 - d On the NNMi management server, run the command `netstat -a`, and then search the output for the port you selected in [step c](#). If that port number does not appear in the output, it is probably available for the HP Operations agent to use.
- 4 On the NNMi management server, configure the HP Operations agent with a custom port for receiving SNMP traps from NNMi by entering the following commands:

- *Windows* NNMi management server:

- Configure the agent:

```
ovconfchg -ns eaagt -set SNMP_TRAP_PORT <custom_port> \
-set SNMP_SESSION_MODE NNM_LIBS
```

- Restart the agent:

```
ovc -restart opctrapi
```

- *UNIX* NNMi management server:

- Configure the agent:

```
ovconfchg -ns eaagt -set SNMP_TRAP_PORT <custom_port> \
-set SNMP_SESSION_MODE NO_TRAPD
```

- Restart the agent:

```
ovc -restart opctrapi
```

For `<custom_port>`, use the port that you identified in [step 3](#) of this procedure.

- 5 On the NNMi management server, configure NNMi incident forwarding to the HP Operations agent:
 - a In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
 - b Click **HPOM agent implementation**, and then click **New**.
(If you have selected an available destination, click **Reset** to make the **New** button available.)
 - c On the **HP NNMi–HPOM Agent Destination** form, select the **Enabled** check box to make the remaining fields on the form available.
 - d Enter the information for connecting to the HP Operations agent on the NNMi management server. The trap destination port is the port that you identified in [step 3](#) of this procedure.

For information about these fields, see [HP Operations Agent Connection](#) on page 502.
 - e Specify the sending options. Select the **HTTP** option for the **NNMi Console Access** field.

For information about these fields, see [HPOM Integration Content](#) on page 503.
 - f Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with the settings, click **Return**, and then adjust the values as suggested by the text of the error message.
- 6 *Optional.* In HPOM, add the custom message attributes for NNMi incidents to the active messages browser. Follow the appropriate steps:
 - *HPOM for Windows:*
 - In the browser, right-click any column heading, and then click **Options**.
 - In the **Enter Custom Message Attributes** list, select an attribute, and then click **Add**.
 - *HPOM for UNIX or HPOM for Linux:*
 - In the Java GUI Message Browser, right-click any column heading, and then click **Customize Message Browser Columns**.
 - On the **Custom** tab, select from the **Available Custom Message Attributes**, and then click **OK**.

Note the following information:

- Most of the custom message attributes for NNMi incidents begin with the text `nmm.`
- For the agent implementation of the HP NNMi–HPOM integration, some interesting attributes for NNMi incidents are as follows:

`nmm.name`
`nmm.server.name`

For information about other interesting CMAs, see [Using the HP NNMi–HPOM Integration \(Agent Implementation\)](#) on page 494.
- To change the order in which the custom message attributes appear in the messages browser, drag a column heading to the new location.

7 *Optional.* On the HPOM management server, enable contextual launching of the NNMi views.

- *HPOM for Windows:* Associate the NNMi source nodes with the HP NNMi Web Tools group.

For information, see *Enable tools in the By Node tool group* in the HPOM help.

- *HPOM for UNIX:* Install the basic set of NNMi applications, and optionally install additional NNMi applications.



HPOM version 9.00 or higher automatically installs the basic NNMi applications.

For information, see the section on installing and configuring the HP NNMi–HPOM integration in the *HP Operations Manager for UNIX Administrator's Reference* (version 9.xx) or the *HP NNMi–HPOM Integration for HP Operations Manager User's Guide* (version 8.3x).

- *HPOM for Linux:* HPOM automatically installs the basic NNMi applications. Optionally install additional NNMi applications.

For information, see the section on installing and configuring the HP NNMi–HPOM integration in the *HP Operations Manager for Linux Administrator's Reference*.

Using the HP NNMi–HPOM Integration (Agent Implementation)

The agent implementation of the HP NNMi–HPOM integration provides a one-way flow of NNMi management events and SNMP traps to the HP Operations agent. The SNMP trap policy conditions determine how HPOM treats and displays the incoming traps. For example, you can change a policy condition to include the value of a trap custom message attribute (CMA) in the message text.



NNMi sends only one copy of each management event or SNMP trap to the HP Operations agent. This behavior is different from that of the NNM 6.x/7.x integration with HPOM.

View the forwarded NNMi incidents in the HPOM active messages browser. HPOM menu commands provide access to NNMi views in the context of the selected message. Information embedded in each message supports this cross-navigation:

- The `nnmi.server.name` and `nnmi.server.port` CMAs in the message identify the NNMi management server.
- The `nnmi.incident.uuid` CMA identifies the incident in the NNMi database.

The original trap source appears in the **Object** column of the HPOM active messages browser and in the `nnm.source.name` CMA. (In the web services implementation of the HP NNMi–HPOM integration, the original trap source is only available in `nnm.source.name` CMA.)

Configuration Item Identifiers

In HP Business Service Management (BSM) and HP Universal CMDB Software (UCMDB), a configuration item (CI) is a database representation of a component in the IT environment. A CI can be a line of business, business process, application, server hardware, or a service.

When NNMi integrates with the BSM topology database or UCMDB, NNMi shares CI information with BSM or UCMDB for the devices that NNMi manages. In this case, the agent implementation of the HP NNMi–HPOM integration can associate incidents regarding NNMi-managed devices with BSM or UCMDB CIs. The SNMP trap policy conditions enable this association.

For information about the integrations with BSM and UCMDB, see:

- [HP Business Service Management Topology](#) on page 415
- [HP Universal CMDB](#) on page 423

Health Indicators

If the NNMi SNMP trap policy file was created with the `-omi_hi` option to `nnmopcexport.ovpl`, the policy file associates a health indicator with each standard NNMi management event in the SNMP trap policy file, as appropriate. (Not all management event types have health indicators.) The health indicator is available in the `EtiHint` CMA.

For the specific health indicators, see the SNMP trap policy file.

Default Policy Conditions

The default integration behavior varies with the integration content, as described here:

- NNMi management event incidents
 - The NNMi SNMP trap policy file includes conditions for all NNMi management event configurations defined in the NNMi incident configuration when the file was generated.
 - The messages created from NNMi management events appear in the HPOM active messages browser.
 - These traps include the CI information described in [Configuration Item Identifiers](#) on page 494.
 - The messages created from these traps might include health indicators described in [Health Indicators](#) on page 495.
- Third-party SNMP traps
 - The NNMi SNMP trap policy file includes conditions for all SNMP trap configurations defined in the NNMi incident configuration when the file was generated.
 - The messages created from third-party traps appear in the HPOM active messages browser.
 - These traps include the CI information described in [Configuration Item Identifiers](#) on page 494.
 - The messages created from these traps do not include health indicators.
 - If you configure the integration to forward all received SNMP traps and the HPOM management server receives SNMP traps directly from devices that NNMi manages, HPOM receives duplicate device traps. You can set the policies to correlate SNMP traps from NNMi with those that HPOM receives directly from managed devices.

- EventLifecycleStateClosed traps
 - The HP Operations agent logs the messages created from these traps. Generally, they do not appear in the HPOM active messages browser.
 - The NNMi SNMP trap policy file causes the HP Operations agent to acknowledge the message that corresponds to the closed NNMi incident in the HPOM active messages browser.
- LifecycleStateChangeEvent traps
 - The NNMi SNMP trap policy file does not include conditions for processing these traps. The HP Operations agent does not forward these traps to the HPOM active messages browser.
- EventDeleted traps
 - The NNMi SNMP trap policy file does not include conditions for processing these traps. The HP Operations agent does not forward these traps to the HPOM active messages browser.
- Correlation notification traps
 - The HP Operations agent logs the messages created from these traps. They do not appear in the HPOM active messages browser.
 - These traps have no impact on the HPOM active messages browser.

Customizing Policy Conditions

To customize the default policy conditions, edit the conditions on the HPOM management server, and then re-deploy the policy to the HP Operations agent on the NNMi management server. For more information, see the following reference:

- *HPOM for Windows: SNMP Interceptor Policies* (version 9.0x) or *Policy development* (version 8.1x) in the HPOM help
- *HPOM for UNIX: HP Operations Manager for UNIX Concepts Guide*
- *HPOM for Linux: HP Operations Manager for Linux Concepts Guide*

More Information

For more information about the agent implementation of the HP NNMi–HPOM integration, see the following references:

- For descriptions of the trap types that the integration sends to the HP Operations agent, see [Using the NNMi Northbound Interface](#) on page 459.
- For information about the format of the traps that NNMi sends to the HP Operations agent, see the `hp-nnmi-nbi.mib` file.
- For detailed information about using the HP NNMi–HPOM integration, see the HPOM documentation.
 - *HPOM for Windows*: See *Agent implementation of the NNMi Adapter* in the HPOM help.
 - *HPOM for UNIX*: See the section on installing and configuring the HP NNMi–HPOM integration in the *HP Operations Manager for UNIX Administrator's Reference* (version 9.xx) or the *HP NNMi–HPOM Integration for HP Operations Manager User's Guide* (version 8.3x).

- *HPOM for Linux*: See the section on installing and configuring the HP NNMi–HPOM integration in the *HP Operations Manager for Linux Administrator's Reference*.

Changing the HP NNMi–HPOM Integration Configuration (Agent Implementation)

Update the SNMP Trap Policy Conditions for New NNMi Traps

If new SNMP trap incident configurations have been added to NNMi since the integration was configured, follow these steps:

- 1 On the NNMi management server, use the `nnmopcexport.ovpl` command to create an SNMP trap policy file for the new traps.

For the `-template` option, specify a name that is different from the names of the existing SNMP trap policy files.

You can limit the file contents to a specific author or OID prefix value. For more information, see the `nnmopcexport.ovpl` reference page, or the UNIX manpage.
- 2 Transfer the new SNMP trap policy file from the NNMi management server to the HPOM management server, and then import it into HPOM.
- 3 On the HPOM management server, deploy the new policy to the NNMi managed node.

Alternatively, you can re-create the SNMP trap policy file for all NNMi management events and SNMP traps. If you take this approach, importing the new policy file into HPOM overwrites any existing policy customizations.

Change the Configuration Parameters

To change the integration configuration parameters, follow these steps:

- 1 In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **HPOM agent implementation**.
- 3 Select a destination, and then click **Edit**.
- 4 Modify the values as appropriate.

For information about the fields on this form, see [HP NNMi–HPOM Agent Destination Form Reference \(Agent Implementation\)](#) on page 502.
- 5 Verify that the **Enable Integration** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

Disabling the HP NNMi–HPOM Integration (Agent Implementation)

No SNMP trap queuing occurs while a destination is disabled.

To discontinue the forwarding of NNMi incidents to the HP Operations agent, follow these steps:

- 1 In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **HPOM agent implementation**.
- 3 Select a destination, and then click **Edit**.
Alternatively, click **Delete** to entirely remove the configuration for the selected destination.
- 4 Clear the **Enable Integration** check box at the top of the form, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

Optionally, deactivate or delete the SNMP trap policy as described in the HPOM documentation.

Troubleshooting the HP NNMi–HPOM Integration (Agent Implementation)

HPOM Active Messages Browser Does Not Receive Any Forwarded Incidents



In the following procedure, the `OVBIN` environment variable refers to the `bin` directory for the HP Operations agent commands, which defaults to the following value:

- *Windows*: `<drive>\Program Files\HP\HP BTO Software\bin`
- *UNIX*: `/opt/OV/bin`

If the HPOM active messages browser does not contain any incidents from NNMi, follow these steps:

- 1 On the NNMi management server, verify the HP Operations agent configuration:
 - *Windows* NNMi management server:
`%OVBIN%\ovconfget eaagt`
 - *UNIX* NNMi management server:
`$OVBIN/ovconfget eaagt`

The command output should include the following information:

- *Windows*: `SNMP_SESSION_MODE=NNM_LIBS`
- *UNIX*: `SNMP_SESSION_MODE=NO_TRAPD`
- `SNMP_TRAP_PORT=<custom_port>`

The value of `<custom_port>` should *not* be 162 and should match the value of the **Port** field on the **HP NNMi–HPOM Agent Destination** form.

- 2 Evaluate the HP Operations agent configuration by considering the results from [step 1](#):
 - If the HP Operations agent configuration is as expected, continue with [step 3](#) of this procedure.
 - If the `SNMP_SESSION_MODE` parameter is not set correctly, repeat [step 4](#) on page 492 until the `ovconfget` command returns the expected results.
 - If the value of `<custom_port>` is 162 or does not match the value of the **Port** field on the **HP NNMi-HPOM Agent Destination** form, repeat [step 3](#) on page 492 through [step 5](#) on page 493, as appropriate, until the `ovconfget` command returns the expected results.

- 3 On the NNMi management server, verify that the HP Operations agent is running:

- *Windows* NNMi management server:
`%OVBIN%\opcagt -status`
- *UNIX* NNMi management server:
`$OVBIN/opcagt -status`

The command output should include an `opctrapi` entry similar to the following example:

```
opctrapi OVO SNMP Trap Interceptor AGENT,EA (4971) Running
```

If the output is not as expected, restart the HP Operations agent:

```
ovc -restart opctrapi
```

- 4 On the NNMi management server, verify that the HP Operations agent is listening on the expected SNMP trap port:

- a Run the following command:

— *Windows*: `netstat -an | findstr <custom_port>`

— *UNIX*: `netstat -an | grep <custom_port>`

Where `<custom_port>` is the value of `SNMP_TRAP_PORT` from [step 1](#) of this procedure.

- b Verify that the output includes the state `LISTENING` or `LISTEN`.

If the output is not as expected, restart the HP Operations agent:

```
ovc -restart opctrapi
```

- 5 On the HPOM management server, verify the external node filter for the NNMi management server node.

The HPOM management server must be configured to accept incidents from the devices that NNMi manages. HPOM ignores any forwarded incident from an NNMi source node that is not configured as a managed node or included in an external node filter, as described in [step 2](#) on page 491.

- 6 On the NNMi management server, verify that the SNMP trap policy file for NNMi has been deployed to the HP Operations agent on the NNMi management server:
 - *Windows* NNMi management server:
`%OVBIN%\ovpolicy -list`
 - *UNIX* NNMi management server:
`$OVBIN/ovpolicy -list`

The command output should include an entry similar to the following example:

Type	Name	Status	Version
trapi	"NNMi Management Events"	enabled	0001.0000

The value of the Name field is the name of the SNMP trap policy file from the `-template` option to `nnmopcexport.ovpl` in [step 1](#) on page 490.

- 7 Verify that the HP Operations agent is receiving traps:
 - a Verify that the HP Operations agent can send messages to the HPOM management server.
 - b Enable tracing of the HP Operations agent to determine whether the traps arrive at the HP Operations agent.

For information about troubleshooting the HP Operations agent, see the following reference:

- *HPOM for Windows: HPOM help*
 - *HPOM for UNIX: HP Operations Manager for UNIX HTTPS Agent Concepts and Configuration Guide*
 - *HPOM for Linux: HP Operations Manager for Linux HTTPS Agent Concepts and Configuration Guide*
- 8 Verify that NNMi is forwarding management events to the HP Operations agent.
 For information, see [Troubleshooting the NNMi Northbound Interface](#) on page 463.

HPOM Active Messages Browser Does Not Receive Some Forwarded Incidents

If one or more NNMi incidents do not appear in the HPOM active messages browser, follow these steps:

- 1 On the NNMi management server, verify that the SNMP trap policy does not suppress the trap.
- 2 On the HPOM management server, verify the external node filter for the NNMi management server node.

The HPOM management server must be configured to accept incidents from the devices that NNMi manages. HPOM ignores any forwarded incident from an NNMi source node that is not configured as a managed node or included in an external node filter, as described in [step 2](#) on page 491.

- 3 On the HPOM management server, verify that HPOM is running.

If the HPOM management server shuts down, the HP Operations agent queues received traps. The HP Operations agent forwards the queued traps when the HPOM management server becomes available.

If the HP Operations agent shuts down, the forwarded traps are lost. NNMi does not resend traps.

- 4 On the NNMi management server, verify that the NNMi processes are running:

```
ovstatus -c
```

Any traps sent to NNMi while it is shut down are lost.

HP NNMi–HPOM Agent Destination Form Reference (Agent Implementation)

The **HP NNMi–HPOM Agent Destination** form contains the parameters for configuring communications between NNMi and the HP Operations agent. This form is available from the **Integration Module Configuration** workspace. (On the **HP NNMi–HPOM Integration Selection** form, click **HPOM agent implementation**. Click **New**, or select a destination, and then click **Edit**.)



Only NNMi users with the Administrator role can access the **HP NNMi–HPOM Agent Destination** form.

The **HP NNMi–HPOM Agent Destination** form collects information for the following areas:

- [HP Operations Agent Connection](#) on page 502
- [HPOM Integration Content](#) on page 503
- [HP Operations Agent Destination Status Information](#) on page 505

To apply changes to the integration configuration, update the values on the **HP NNMi–HPOM Agent Destination** form, and then click **Submit**.

HP Operations Agent Connection

[Table 52](#) lists the parameters for configuring the connection to the HP Operations agent.

Table 52 HP Operations Agent Connection Information

Field	Description
Host	<p>The fully-qualified domain name (preferred) or the IP address of the NNMi management server, which is the system on which the HP Operations agent receives SNMP traps from NNMi.</p> <p>The integration supports the following methods for identifying the HP Operations agent host:</p> <ul style="list-style-type: none">• NNMi FQDN NNMi manages the connection to the HP Operations agent on the NNMi management server and the Host field becomes read-only. This is the default and recommended configuration.• Use Loopback Do not use this option.• Other Do not use this option. <p>NOTE: If the NNMi management server participates in NNMi application failover, see Application Failover and the NNMi Northbound Interface on page 465 for information about the impact of application failover on the integration module.</p>

Table 52 HP Operations Agent Connection Information (cont'd)

Field	Description
Port	<p>The UDP port where the HP Operations agent receives SNMP traps.</p> <p>Enter the port number specific to the HP Operations agent. This value is the port that you identified in step 3 on page 492.</p> <p>To determine the port, run the <code>ovconfget -eaagt</code> command on the NNMi management server. The trap port is the value of the <code>SNMP_TRAP_PORT</code> variable.</p> <p>NOTE: This port number must be different from the port on which NNMi receives SNMP traps, as set in the SNMP Port field on the Communication Configuration form in the NNMi console.</p>
Community String	<p>A read-only community string for the HP Operations agent to receive traps.</p> <p>For the HP NNMi—HPOM integration, use the default value, which is <code>public</code>.</p>

HPOM Integration Content

[Table 53](#) lists the parameters for configuring which content NNMi sends to the HP Operations agent.

Table 53 HPOM Integration Content Configuration Information

Field	Description
Incidents	<p>The incident forwarding specification.</p> <ul style="list-style-type: none"> • Management NNMi forwards only NNMi-generated management events to the HP Operations agent. • SNMP 3rd Party Trap NNMi forwards only SNMP traps that NNMi receives from managed devices to the HP Operations agent. • Both NNMi forwards to the HP Operations agent both NNMi-generated management events and SNMP traps that NNMi receives from managed devices. This is the default configuration. <p>NNMi begins forwarding incidents as soon as you enable the destination.</p> <p>For more information, see Incident Forwarding on page 459.</p>

Table 53 HPOM Integration Content Configuration Information (cont'd)

Field	Description
Lifecycle State Changes	<p>The incident change notification specification.</p> <ul style="list-style-type: none"> • Enhanced Closed NNMi sends an incident closed trap to the HP Operations agent for each incident that changes to the CLOSED lifecycle state. This is the default configuration. • State Changed NNMi sends an incident lifecycle state changed trap to the HP Operations agent for each incident that changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state. • Both NNMi sends an incident closed trap to the HP Operations agent for each incident that changes to the CLOSED lifecycle state. Additionally, the integration sends an incident lifecycle state changed trap to the HP Operations agent for each incident that changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state. NOTE: In this case, each time an incident changes to the CLOSED lifecycle state, the integration sends two notification traps: an incident closed trap and an incident lifecycle state changed trap. <p>For more information, see Incident Lifecycle State Change Notifications on page 460.</p>
Correlations	<p>The incident correlation notification specification.</p> <ul style="list-style-type: none"> • None NNMi does not notify the HP Operations agent of incident correlations resulting from NNMi causal analysis. This is the default configuration. • Single NNMi sends a trap for each parent-child incident correlation relationship resulting from NNMi causal analysis. • Group NNMi sends one trap per correlation that lists all child incidents correlated to a parent incident. <p>For more information, see Incident Correlation Notifications on page 461.</p>
Deletions	<p>The incident deletion specification.</p> <ul style="list-style-type: none"> • Don't Send NNMi does not notify the HP Operations agent when incidents are deleted in NNMi. This is the default configuration. • Send NNMi sends a deletion trap to the HP Operations agent for each incident that is deleted in NNMi. <p>For more information, see Incident Deletion Notifications on page 462.</p>
NNMi Console Access	<p>The connection protocol specification in the URL for browsing to the NNMi console from the HPOM message browser. The traps that NNMi sends to the HP Operations agent include the NNMi URL in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2). The integration requires an HTTP connection to the NNMi console. Select the HTTP option.</p>

Table 53 HPOM Integration Content Configuration Information (cont'd)

Field	Description
Incident Filters	<p>A list of object identifiers (OIDs) on which the integration filters the events sent to the HP Operations agent. Each filter entry can be a valid numeric OID (for example, .1.3.6.1.6.3.1.1.5.4.1.3.6.1.4.1.9) or OID prefix (for example, .1.3.6.1.6.3.1.1.5.*).</p> <p>Select one of the following options:</p> <ul style="list-style-type: none"> • None NNMi sends all events to the HP Operations agent. This is the default configuration. • Include NNMi sends only the specific events that match the OIDs identified in the filter. • Exclude NNMi sends all events except for the specific events that match the OIDs identified in the filter. <p>Specify the incident filter:</p> <ul style="list-style-type: none"> • To add a filter entry, enter the text in the lower text box, and then click Add. • To delete a filter entry, select that entry from the list in the upper box, and then click Remove. <p>For more information, see Event Forwarding Filter on page 462.</p>

HP Operations Agent Destination Status Information

Table 54 lists the read-only status information for the HP Operations agent. This information is useful for verifying that the integration is working correctly.

Table 54 HP Operations Agent Destination Status Information

Field	Description
Trap Destination IP Address	The IP address to which the HP Operations agent destination host name resolves. This value is unique to this HP Operations agent destination.
Uptime (seconds)	<p>The time (in seconds) since the northbound component was last started. The traps that NNMi sends to the HP Operations agent include this value in the sysUptime field (1.3.6.1.2.1.1.3.0).</p> <p>This value is the same for all integrations that use the NNMi northbound interface. To see the latest value, either refresh or close and re-open the form.</p>
NNMi URL	<p>The URL for connecting to the NNMi console. The traps that NNMi sends to the HP Operations agent include this value in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2).</p> <p>This value is unique to this northbound destination.</p>

HP NNMi—HPOM Integration (Web Services Implementation)

The agent implementation of the HP NNMi—HPOM integration is the preferred solution for integrating HPOM with NNMi.

If the agent and the web services implementations of the HP NNMi—HPOM integration both forward messages to the same HPOM management server, you might not see all messages from both implementations in the HPOM active messages browser. For this reason, HP does not support running both implementations of the HP NNMi—HPOM integration from one NNMi management server to the same HPOM management server concurrently.

This section contains the following topics:

- [About the HP NNMi—HPOM Integration \(Web Services Implementation\)](#) on page 506
- [Enabling the HP NNMi—HPOM Integration \(Web Services Implementation\)](#) on page 508
- [Using the HP NNMi—HPOM Integration \(Web Services Implementation\)](#) on page 512
- [Changing the HP NNMi—HPOM Integration Configuration \(Web Services Implementation\)](#) on page 513
- [Disabling the HP NNMi—HPOM Integration \(Web Services Implementation\)](#) on page 514
- [Troubleshooting the HP NNMi—HPOM Integration \(Web Services Implementation\)](#) on page 514
- [HP NNMi—HPOM Web Services Integration Configuration Form Reference](#) on page 519

About the HP NNMi—HPOM Integration (Web Services Implementation)

The web services implementation of the HP NNMi—HPOM integration forwards NNMi incidents to the HPOM active messages browser. The integration synchronizes incidents between NNMi and HPOM. It also provides for accessing the NNMi console from within HPOM.

The HP NNMi—HPOM integration supports a “many-to-many” arrangement. Each NNMi management server can forward incidents to multiple HPOM management servers. Likewise, each HPOM management server can receive incidents from multiple NNMi management servers. The integration interprets the unique identifier of an incident to determine the source NNMi management server.

The HP NNMi—HPOM integration consists of the following components:

- **HP NNMi—HPOM Integration Module**

The HP NNMi—HPOM integration module forwards incidents from NNMi to HPOM. It is installed and configured on the NNMi management server.
- **HP Operations Manager Incident Web Service**

HPOM uses the HP Operations Manager Incident Web Service (IWS) to receive the incidents that are forwarded from NNMi.

- **HPOM applications for contextual access of the NNMi console**

HPOM provides applications for accessing forms, views, and tools in the NNMi console. For example, you can open an NNMi incident directly from the HPOM active messages browser. The specific application determines the context in which the NNMi console opens. You need to configure the applications before you can use them.

Value

The HP NNMi–HPOM integration provides event consolidation in the HPOM active messages browser for the network management, system management, and application management domains, so that HPOM users can detect and investigate potential network problems.

The primary features of the integration are as follows:

- Automatic incident forwarding from NNMi to HPOM.
 - Forwarded incidents appear in the HPOM active messages browser.
 - You can create filters that limit which incidents NNMi forwards.
- Synchronization of Incident updates between NNMi and HPOM as described in the following table.

Trigger	Result
In HPOM, the message is acknowledged.	In NNMi, the corresponding incident's lifecycle state is set to Closed.
In HPOM, the message is unacknowledged.	In NNMi, the corresponding incident's lifecycle state is set to Registered.
In NNMi, the incident's lifecycle state is set to Closed.	In HPOM, the corresponding message is acknowledged.
In NNMi, the incident's lifecycle state is changed from Closed to any other state.	In HPOM, the corresponding message is unacknowledged.

- Access to the NNMi console from HPOM.
 - HPOM users can open the NNMi **Incident** form in the context of a selected message.
 - HPOM users can launch an NNMi view (for example, the Layer 2 Neighbor view) in the context of a selected message and node.
 - HPOM users can launch an NNMi tool (for example, status poll) in the context of a selected message and node.
 - When HPOM is consolidating NNMi incidents from multiple NNMi management servers, the integration interprets the unique identifier of each incident to access the correct NNMi management server.

Supported Versions

The information in this section applies to the following product versions:

- HPOM for Windows version 8.10 or higher
- HPOM for UNIX version 8.30 or higher (including HPOM for UNIX version 9.00 or higher)
- HPOM for Linux version 9.00 or higher
- NNMi version 9.00 or higher

NNMi and HPOM must be installed on separate computers. The NNMi management server and the HPOM management server computer can be of the same or different operating systems.

For the most recent information about supported hardware platforms and operating systems, see the support matrices for both products.

Documentation

This chapter describes how to configure NNMi to communicate with HPOM.

The HPOM documentation describes how to configure HPOM to communicate with NNMi. It also describes how to use the HP NNMi–HPOM integration.

- For HPOM for Windows, see the information for the HP NNMi Adapter in the HPOM help.
- For HPOM for UNIX version 9.xx, see the *Integrating NNMi into HPOM* section in the *HP Operations Manager for UNIX Administrator's Reference*.
- For HPOM for UNIX version 8.3x, see the *HP NNMi–HPOM Integration for HP Operations Manager User's Guide*.
- For HPOM for Linux, see the *Integrating NNMi into HPOM* section in the *HP Operations Manager for Linux Administrator's Reference*.

Enabling the HP NNMi–HPOM Integration (Web Services Implementation)

This section describes the procedure for enabling the HP NNMi–HPOM integration. For each NNMi management server and each HPOM management server that you want to include in the integration, complete the appropriate steps in the procedure for the version of HPOM that you are using.

HPOM for Windows

- 1 On the NNMi management server, configure NNMi incident forwarding to HPOM:
 - a In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
 - b Click **web services implementation**.
 - c On the **HP NNMi–HPOM Web Services Integration Configuration** form, select the **Enable Integration** check box to make the remaining fields on the form available.



- d Enter the information for connecting to the NNMi management server.

The integration requires an HTTP connection to the NNMi console. Leave the **NNMi SSL Enabled** check box cleared.

For information about these fields, see [NNMi Management Server Connection](#) on page 519.

- e Enter the information for connecting to the HPOM management server.

For information about these fields, see [HPOM Management Server Connection](#) on page 520.

- f Enter values for the following fields:

- **Forward Only**
- **Holding period (minutes)**
- **Incident Filter**

For information about these fields, see [Integration Behavior](#) on page 521.

- g If you want NNMi to forward incidents to multiple HPOM management servers, click **Add another HPOM server**, and then enter the information for the next HPOM management server in the HPOM fields.

The information for the first server appears in the **Additional HPOM Servers** list.

- h Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with connecting to the HPOM server, re-open the **HP NNMi–HPOM Web Services Integration Configuration** form (or press **ALT+LEFT ARROW** in the message window), and then adjust the values for connecting to the HPOM management server as suggested by the text of the error message.

- 2 In HPOM, configure the NNMi adapter for connecting to the NNMi management server as described in *Configure the NNMi Server Name and Port* of the HPOM help.
- 3 In HPOM, add a managed node for each NNMi node that will be named as a source node in the NNMi incidents that are forwarded to this HPOM management server. Also add a managed node for each NNMi management server that will forward incidents to this HPOM management server.

Alternatively, you can create one external node to catch all forwarded NNMi incidents. For initial testing, set the node filter to `<*>.<*>.<*>.<*>` (for an IP filter) or `<*>` (for a name filter). After you validate the integration, restrict the external node filter to match your network.

For more information, see *Configuring NNMi Server Nodes* in the HPOM help.




If you do not set up an HPOM managed node for an NNMi incident source node, the HPOM management server discards all incidents regarding that node.

- 4 *Optional.* In HPOM, add the custom message attributes for NNMi incidents to the active messages browser:
 - a In the browser, right-click any column heading, and then click **Options**.
 - b In the **Enter Custom Message Attributes** list, select an attribute, and then click **Add**.
 - The custom message attributes for NNMi incidents begin with the text `nmm`.
 - For the web services implementation of the HP NNMi–HPOM integration, the most interesting attributes for NNMi incidents are as follows:


```
nmm.assignedTo
nmm.category
nmm.emittingNode.name
nmm.source.name
```
 - To change the order in which the custom message attributes appear in the messages browser, drag a column heading to the new location.
- 5 *Optional.* In HPOM, enable contextual launching of the NNMi views by associating the NNMi source nodes with the HP NNMi Web Tools group.
For more information, see *Enable tools in the By Node tool group* in the HPOM help.

HPOM for UNIX and HPOM for Linux

- 1 *HPOM for UNIX version 8.3x only.* Prepare the HPOM for UNIX management server:
 - a On the HPOM for UNIX management server, install the HP Operations Manager Incident Web Service (IWS) as described in the *HP Operations Manager Incident Web Service Integration Guide*.
 - b On the HPOM for UNIX management server, install the most recent HPOM consolidated patch, which is available at:
<http://h20230.www2.hp.com/selfsolve/patches>
- 2 On the NNMi management server, configure NNMi incident forwarding to HPOM:
 - a In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
 - b Click **web services implementation**.
 - c On the **HP NNMi–HPOM Web Services Integration Configuration** form, select the **Enable Integration** check box to make the remaining fields on the form available.
 - d Enter the information for connecting to the NNMi management server.
 The integration requires an HTTP connection to the NNMi console. Leave the **NNMi SSL Enabled** check box cleared.
For information about these fields, see [NNMi Management Server Connection](#) on page 519.
 - e Enter the information for connecting to the HPOM management server.
For information about these fields, see [HPOM Management Server Connection](#) on page 520.

f Enter values for the following fields:

- **Forward Only**
- **Holding period (minutes)**
- **Incident Filter**

For information about these fields, see [Integration Behavior](#) on page 521.

g If you want NNMi to forward incidents to multiple HPOM management servers, click **Add another HPOM server**, and then enter the information for the next HPOM management server in the HPOM fields.

The information for the first server appears in the **Additional HPOM Servers** list.

h Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with connecting to the HPOM server, re-open the **HP NNMi–HPOM Web Services Integration Configuration** form (or press **ALT+LEFT ARROW** in the message window), and then adjust the values for connecting to the HPOM management server as suggested by the text of the error message.

i Click **Submit** at the bottom of the form.

3 In HPOM, add a managed node for each NNMi node that will be named as a source node in the NNMi incidents that are forwarded to this HPOM management server. Also add a managed node for each NNMi management server that will forward incidents to this HPOM management server.

Alternatively, you can create one external node to catch all forwarded NNMi incidents. For initial testing, set the node filter to `<*>.<*>.<*>.<*>` (for an IP filter) or `<*>` (for a name filter). After you validate the integration, restrict the external node filter to match your network.

For more information, see the *HP Operations Manager for UNIX Administrator's Reference* or the *HP Operations Manager for Linux Administrator's Reference*.



If you do not set up an HPOM managed node for an NNMi incident source node, the HPOM management server discards all incidents regarding that node.

4 *Optional.* In HPOM, add the custom message attributes for NNMi incidents to the active messages browser:

a In the Java GUI Message Browser, right-click any column heading, and then click **Customize Message Browser Columns**.

b On the **Custom** tab, select from the **Available Custom Message Attributes**, and then click **OK**.

— The custom message attributes for NNMi incidents begin with the text `nnm.`

— For the web services implementation of the HP NNMi–HPOM integration, the most interesting attributes for NNMi incidents are as follows:

```
nnm.assignedTo
nnm.category
nnm.emittingNode.name
nnm.source.name
```

— To change the order in which the custom message attributes appear in the messages browser, drag a column heading to the new location.

- 5 *Optional.* On the HPOM management server, prepare the HPOM applications for accessing the NNMi console.
 - a *Required.* Install the basic set of NNMi applications.
 - HPOM version 9.00 or higher automatically installs the basic NNMi applications.
 - b *Optional.* Install additional NNMi applications.

For information, see the section on installing and configuring the HP NNMi–HPOM integration in the *HP Operations Manager for UNIX Administrator's Reference* (version 9.xx), the *HP NNMi–HPOM Integration for HP Operations Manager User's Guide* (version 8.3x), or the *HP Operations Manager for Linux Administrator's Reference*.

Using the HP NNMi–HPOM Integration (Web Services Implementation)

Usage Example

Figure 21 shows an interface down incident in the NNMi console. The information in the **Source Object** and **Message** columns together describe the situation.

Figure 21 Interface Down Incident in NNMi Console

Se	LS	Last Occurren	Source Node	Source Objec	Ca	Fa	CN	Message
		9/25/08 4:29 PM	ovcrt1	Et1/0				Cisco Agent Interface Down (linkDown Trap) on

Figure 22 shows the NNMi incident as received by HPOM for Windows. Figure 23 shows the NNMi incident as received by HPOM for UNIX. The **nnm.source.name** and **Text** columns are equivalent to the **Source Object** and **Message** columns in the NNMi console.

- ▶ You must enable the display of the **nnm.source.name** custom message attribute column as described in [step 4](#) on page 510 (for HPOM for Windows) and in [step 4](#) on page 511 (for HPOM for UNIX and for HPOM for Linux).

Figure 22 Forwarded Incident in HPOM for Windows

Severity	Received	Node	Application	Object	Text	nnm.source.name
Critical	26/08/2008 16:2...	ovcrt1....	NNMi	Interface	Cisco Agent Interface Down (linkDo...	Et1/0

Figure 23 Forwarded Incident in HPOM for UNIX

Severity	Time Received	Node	Application	Object	Message Text	nnm.source.name
Critical	08:56:39 09/2...	ovcrt1....	NNMi	Interface	Cisco Agent Interface Down (linkDown Trap) on interf...	Et1/0

A Normal Situation: Unknown MSI Condition

The HPOM server receives forwarded NNMi incidents through MSI (not a regular trap policy). In the HPOM message browser, the format of the message source is **MSI** followed by the name of the MSI interface. The condition name corresponds to the `condition_id` field in the message, which is unset because there is no associated policy.

- *HPOM for Windows*: The policy type is empty.
- *HPOM for UNIX* or *HPOM for Linux*: The message source is of the format: **MSI: <MSI_Interface>: Unknown Condition.**

More Information

For detailed information about using the HP NNMi–HPOM integration, see the HPOM documentation.

- *HPOM for Windows*: See the topics about the HP NNMi adapter in the HPOM help.
- *HPOM for UNIX*: See the section on installing and configuring the HP NNMi–HPOM integration in the *HP Operations Manager for UNIX Administrator's Reference* (version 9.xx) or the *HP NNMi–HPOM Integration for HP Operations Manager User's Guide* (version 8.3x).
- *HPOM for Linux*: See the section on installing and configuring the HP NNMi–HPOM integration in the *HP Operations Manager for Linux Administrator's Reference*.



In the HPOM messages browser, the details for a forwarded NNMi incident are available as custom message attributes.

Changing the HP NNMi–HPOM Integration Configuration (Web Services Implementation)

- 1 In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **web services implementation**.
- 3 Modify the values as appropriate.
 - If you know the syntax of the entries in the Incident Filter and Additional HPOM Servers lists, you can modify the entries directly.
 - If you do not know the syntax for a list item, delete that entry and then re-enter it.

For information about the fields on this form, see [HP NNMi–HPOM Web Services Integration Configuration Form Reference](#) on page 519.

- 4 Verify that the **Enable Integration** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

Disabling the HP NNMi–HPOM Integration (Web Services Implementation)

For All HPOM Management Servers

To discontinue the forwarding of NNMi incidents to all HPOM management servers, follow these steps:

- 1 In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **web services implementation**.
- 3 Clear the **Enable Integration** check box at the top of the form, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

If necessary, repeat this process for all NNMi management servers.

For One HPOM Management Server

To discontinue the forwarding of NNMi incidents to only one of the HPOM management servers, follow these steps:

- 1 In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **web services implementation**.
- 3 In the **Additional HPOM Servers** list, edit the text to delete the entry (or entries) for the HPOM management server to disconnect from the integration.



Clicking **Clear** removes all HPOM servers from the list.

- 4 Click **Submit** at the bottom of the form.

The changes take effect immediately.

Troubleshooting the HP NNMi–HPOM Integration (Web Services Implementation)

HPOM Does Not Receive Any Forwarded Incidents



If the integration has worked successfully in the past, it is possible that some aspect of the configuration, for example, the NNMi or HPOM user password, has changed recently. You may want to update the integration configuration as described in [Changing the HP NNMi–HPOM Integration Configuration \(Web Services Implementation\)](#) on page 513, before walking through this entire procedure.

- 1 In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **web services implementation**.

For information about the fields on this form, see [HP NNMi–HPOM Web Services Integration Configuration Form Reference](#) on page 519.

- 3 Check the status of the integration, in the **HP NNMi–HPOM Web Services Integration Configuration** form, by clicking **Submit** at the bottom of the form (without making any configuration changes).

A new window displays a status message.

- If the message indicates success, the problem is most likely that HPOM is not configured to accept incidents from the devices that NNMi manages. HPOM ignores any forwarded incident from an NNMi source node that is not configured as a managed node in HPOM, as described in [step 3](#) on page 509 (for HPOM for Windows) and in [step 3](#) on page 511 (for HPOM for UNIX and for HPOM for Linux). Verify the HPOM configuration, and then test the integration as described in [step 10](#) of this procedure.
 - If the message indicates a problem with connecting to the HPOM server, NNMi and HPOM are not able to communicate. Continue with [step 4](#) of this procedure.
- 4 Verify the accuracy and access level of the HPOM credentials by logging in to the HPOM console and displaying the HPOM active messages browser:

- *HPOM for Windows*: Log on to the computer as the **HPOM User** from the **HP NNMi–HPOM Web Services Integration Configuration** form, and then start the HPOM console.

The user name is in the format `<Windows_domain>\<username>`.

- *HPOM for UNIX or HPOM for Linux*: Log on to the HPOM console with the credentials for the **HPOM User** from the **HP NNMi–HPOM Web Services Integration Configuration** form.

If you cannot log on to the HPOM console, contact the HPOM administrator to verify your logon credentials.

- 5 Verify that the connection to the HPOM management server is configured correctly:

- a In a web browser, enter the following URL:

`<protocol>://<omserver>:<port>/opr-webservice//Incident.svc?wsdl`

Where the variables are related to values on the **HP NNMi–HPOM Web Services Integration Configuration** form as follows:

- If the **HPOM SSL Enabled** check box is selected, `<protocol>` is `https`.
- If the **HPOM SSL Enabled** check box is cleared, `<protocol>` is `http`.
- `<omserver>` is the value of **HPOM Host**.
- `<port>` is the value of **HPOM Port**.

- b When prompted, enter the credentials for the **HPOM User** from the **HP NNMi–HPOM Web Services Integration Configuration** form.

The resulting web page is an XML file that describes the IWS.

- If the XML file appears, the connection to the HPOM management server is configured correctly. Continue with [step 6](#).
- If you see an error message, the connection to the HPOM management server is not configured correctly. Contact the HPOM administrator to verify the information that you are using to connect to the HPOM web service. Continue to troubleshoot the connection to HPOM until you see the XML file.

- 6 Verify that the connection to NNMi is configured correctly:



If you used the information described in this step to connect to the NNMi console in [step 1](#) of this procedure, you do not need to reconnect to the NNMi console. Continue with [step 7](#).

- a In a web browser, enter the following URL:

<protocol>://<NNMi server>:<port>/nmm/

Where the variables are related to values on the **HP NNMi–HPOM Web Services Integration Configuration** form as follows:

- If the **NNMi SSL Enabled** check box is selected, *<protocol>* is `https`.



If the **NNMi SSL Enabled** check box is selected, verify that the KeyManager process is running by entering the following command:

```
ovstatus -v ovjboss
```

- If the **NNMi SSL Enabled** check box is cleared, *<protocol>* is `http`.

- *<NNMi server>* is the value of **NNMi Host**.



Use the fully-qualified domain name or the IP address of the NNMi management server. Do not use `localhost`.

- *<port>* is the value of **NNMi Port**.



To verify the NNMi ports for HTTP or HTTPS, check the `nms-local.properties` file, as described in [Table 55](#) on page 519.

- b When prompted, enter the credentials for an NNMi user with the Administrator role.

You should see the NNMi console. If the NNMi console does not appear, contact the NNMi administrator to verify the information that you are using to connect to NNMi. Continue to troubleshoot the connection to NNMi until the NNMi console appears.



You cannot sign in to the NNMi console as a user with the Web Service Client role.

- c Verify the values of the **NNMi User** and **NNMi Password**.

- If the **NNMi User** listed on the **HP NNMi–HPOM Web Services Integration Configuration** form has the Administrator role and you were able to connect to the NNMi console with this user name, then re-enter the corresponding password on the **HP NNMi–HPOM Web Services Integration Configuration** form.

- If the **NNMi User** listed on the **HP NNMi–HPOM Web Services Integration Configuration** form has the Web Service Client role, contact the NNMi administrator to verify the values of **NNMi User** and **NNMi Password**.

Passwords are hidden in the NNMi console. If you are not sure what password to specify for an NNMi user name, ask the NNMi administrator to reset the password.

- 7 Update the **HP NNMi–HPOM Web Services Integration Configuration** form with the values that you used for successful connections in [step 5](#) and [step 6](#) of this procedure.

For more information, see [HP NNMi–HPOM Web Services Integration Configuration Form Reference](#) on page 519.

- 8 Click **Submit** at the bottom of the form.
- 9 If the status message still indicates a problem with connecting to the HPOM server, do the following:
 - a Clear the web browser cache.
 - b Clear all saved form or password data from the web browser.
 - c Close the web browser window completely, and then re-open it.
 - d Repeat [step 7](#) and [step 8](#) of this procedure.
- 10 Test the configuration by generating an incident on the NNMi management server and determining whether it reaches the HPOM management server.

Alternatively, change the lifecycle state of an NNMi management event to OPEN. (If the lifecycle state is currently OPEN, change the lifecycle state to CLOSED and then back to OPEN.)

HPOM Does Not Receive Some Forwarded Incidents

Verify the HPOM nodes and the incident filter.

The HPOM management server must be configured to accept incidents from the devices that NNMi manages. HPOM ignores any forwarded incident from an NNMi source node that is not configured as a managed node in HPOM, as described in [step 3](#) on page 509 (for HPOM for Windows) and in [step 3](#) on page 511 (for HPOM for UNIX and for HPOM for Linux).

If the NNMi source node is configured as a managed node in HPOM, verify the incident filter configuration on the **HP NNMi–HPOM Web Services Integration Configuration** form. Then test the filter by generating an incident on the NNMi management server and determine whether it reaches the HPOM management server.

NNMi Incident Information Is Not Available in the HPOM Messages Browser

The important information from NNMi incidents is passed to HPOM as custom message attributes. Add one or more custom messages attributes for NNMi incidents as described in [step 4](#) on page 510 (for HPOM for Windows) and in [step 4](#) on page 511 (for HPOM for UNIX and for HPOM for Linux).

NNMi and HPOM Are Not Synchronized

If either of the management servers becomes unreachable, the incidents in the NNMi incident views and the HPOM active messages browser might become mismatched. The HP NNMi–HPOM integration can re-synchronize the incidents as described here.

- If an HPOM management server becomes unavailable to the HP NNMi–HPOM integration module, the integration module periodically checks for the availability of that HPOM management server and resumes incident forwarding when a

connection can be re-established. When the connection to the HPOM management server is available, the integration module forwards any incidents that might have been missed while the HPOM management server was down.

- If the NNMi management server is unavailable when an HPOM user acknowledges or unacknowledges a forwarded incident, NNMi does not receive the change of state. NNMi and HPOM might show different states for this incident.

The Integration Does Not Work Through a Firewall

Ensure that the NNMi management server is able to directly address the HPOM IWS by host and port.

HP NNMI–HPOM Web Services Integration Configuration Form Reference

The **HP NNMI–HPOM Web Services Integration Configuration** form contains the parameters for configuring communications between NNMI and HPOM. This form is available from the **Integration Module Configuration** workspace. (On the **HP NNMI–HPOM Integration Selection** form, click **web services implementation**.)



Only NNMI users with the Administrator role can access the **HP NNMI–HPOM Web Services Integration Configuration** form.

The **HP NNMI–HPOM Web Services Integration Configuration** form collects information for the following general areas:

- [NNMI Management Server Connection](#) on page 519
- [HPOM Management Server Connection](#) on page 520
- [Integration Behavior](#) on page 521
- [Incident Filters](#) on page 522

To apply changes to the integration configuration, update the values on the **HP NNMI–HPOM Web Services Integration Configuration** form, and then click **Submit**.

NNMI Management Server Connection

[Table 55](#) lists the parameters for connecting to the NNMI management server. This is the same information that you use to open the NNMI console. You can determine many of these values by examining the URL that invokes an NNMI console session. Coordinate with the NNMI administrator to determine the appropriate values for this section of the configuration form.

Table 55 NNMI Management Server Connection Information

Field	Description
NNMI SSL Enabled	The connection protocol specification for connecting to the NNMI console. The integration requires an HTTP connection to the NNMI console. Leave the NNMI SSL Enabled check box cleared.
NNMI Host	The fully-qualified domain name of the NNMI management server. This field is pre-filled with the host name that was used to access the NNMI console. Verify that this value is the name that is returned by the <code>nnmofficialfqdn.ovpl -t</code> command run on the NNMI management server.
NNMI Port	The port for connecting to the NNMI console. This field is pre-filled with the port that the jboss application server uses for communicating with the NNMI console, as specified in the following file: <ul style="list-style-type: none">• <i>Windows</i>: %NnmDataDir%\conf\nnm\props\nms-local.properties• <i>UNIX</i>: \$NnmDataDir/conf/nnm/props/nms-local.properties For non-SSL connections, use the value of <code>jboss.http.port</code> , which is 80 or 8004 by default (depending on the presence of another web server when NNMI was installed).

Table 55 NNMi Management Server Connection Information (cont'd)

Field	Description
NNMi User	The user name for connecting to the NNMi console. This user must have the NNMi Administrator or Web Service Client role. Best practice: Create and use an <code>Integration</code> user account with the Web Service Client role.
NNMi Password	The password for the specified NNMi user.

HPOM Management Server Connection

Table 56 lists the parameters for connecting to the web service on the HPOM management server. Coordinate with the HPOM administrator to determine the appropriate values for this section of the configuration.

Table 56 HPOM Management Server Connection Information

HPOM Server Parameter	Description
HPOM SSL Enabled	The connection protocol specification. <ul style="list-style-type: none"> If HPOM is configured to use HTTPS, select the HPOM SSL Enabled check box. This is the default configuration. If HPOM is configured to use HTTP, clear the HPOM SSL Enabled check box.
HPOM Host	The fully-qualified domain name of the HPOM management server. Verify that this name is resolvable from the NNMi management server by using the <code>nslookup</code> or <code>ping</code> command. If DNS is questionable, use the IP address of the HPOM management server. If possible, use the <code>tracert</code> command to verify the network path from the NNMi management server to the HPOM management server.
HPOM Port	The port for connecting to the HPOM web service. To determine which port number to specify, do the following on the HPOM management server: <ul style="list-style-type: none"> <i>HPOM for Windows</i>: Examine the port settings in the IIS Manager, which is available from the Start menu, for example, Start > Administrative Tools > Internet Information Services (IIS) Manager. <i>HPOM for UNIX</i> or <i>HPOM for Linux</i>: Run the following command: <code>ovtomcatbct1 -getconf</code> This field is pre-filled with the value 443, which is the default port for SSL connections to HPOM for Windows. For SSL connections to HPOM for UNIX or HPOM for Linux, the default port is 8443 or 8444.

Table 56 HPOM Management Server Connection Information (cont'd)

HPOM Server Parameter	Description
HPOM User	<p>A valid HPOM user account name with the HPOM Administrator role. This user must be permitted to view the HPOM active messages browser and the HPOM incident web service WSDL.</p> <p><i>Windows only:</i> On the Windows operating system, HPOM works through Microsoft Internet Information Services (IIS) to authenticate user credentials. Specify a Windows user in the format <code><Windows_domain>\<username></code>.</p> <p>Best Practice:</p> <ul style="list-style-type: none"> • <i>HPOM for Windows:</i> Specify a user who is a member of the HP-OVE-ADMINS user group. (Verify group membership in the Local Users and Groups area of the Microsoft Management Console, which is available from Control Panel > Administrative Tools > Computer Management.) • <i>HPOM for UNIX or HPOM for Linux:</i> Use the <code>opc_adm</code> user account.
HPOM Password	The password for the specified HPOM user.

Integration Behavior

Table 57 lists the parameters that describe the integration behavior. Coordinate with the NNMi administrator to determine the appropriate values for this section of the configuration.

Table 57 Integration Behavior Information

Field	Description
Forward Only	<p>The behavior specification for the HP NNMi–HPOM integration module. By default, the integration module forwards incidents to and receives incident acknowledgements from the HPOM management servers identified on the HP NNMi–HPOM Web Services Integration Configuration form. You can disable the receipt of incident acknowledgements.</p> <ul style="list-style-type: none"> • For one-way communication (forward incidents to HPOM but ignore incident acknowledgements from HPOM), select the Forward Only check box. • For two-way communication, leave the Forward Only check box cleared. This is the default behavior.
Holding period (minutes)	<p>The number of minutes to wait before forwarding the configured incidents to HPOM. If an incident is closed during this time (for example, an SNMPLinkUp incident cancels an SNMPLinkDown incident), HPOM never receives that incident. If you want NNMi to forward incidents immediately, enter the value 0.</p> <p>The default value is 5 minutes.</p>
Incident Filter	<p>A filter based on NNMi incident attributes that limits incident forwarding. The default filter (<code>nature=ROOTCAUSE origin=MANAGEMENTSOFTWARE</code>) specifies all root cause incidents that are generated by NNMi. You can modify the filter to change which incidents are forwarded to HPOM.</p> <p>NOTE: All text (attribute names and values) in the Incident Filter field is case-sensitive.</p> <p>For more information, see Incident Filters.</p>

Incident Filters

The incident filter is the combination of all entries in the **Incident Filter** list. Filter entries with the same attribute value expand the filter (logical OR). Filter entries with different attribute values restrict the filter (logical AND). All filter entries work together; you *cannot* create a filter of the format (a AND b) OR c. For example filter entries, see [Example Incident Filters](#) on page 523.

To create the incident filter, follow these steps:

- 1 In the NNMi console, open the **HP NNMi–HPOM Integration Selection** form (**Integration Module Configuration > HPOM**).
- 2 Click **web services implementation**.
- 3 To delete a filter entry, in the **Incident Filter** list, edit the text to delete the entry (or entries).



Clicking **Clear** removes all filter entries from the list.

- 4 To add an incident filter entry:
 - a Select an attribute from the **name** list. For the supported attributes, see the table in [step c](#).
 - b Select the comparison operation to perform. Supported operators are:
 - =
 - !=
 - <
 - <=
 - >
 - >=
 - c Enter a comparison value. The following table lists the supported attributes and the acceptable values for each attribute.

Attribute	Possible Values
name	Examine the incident configuration in the NNMi console to determine the available incident names.
nature	<ul style="list-style-type: none">• ROOTCAUSE• SECONDARYROOTCAUSE• SYMPTOM• SERVICEIMPACT• STREAMCORRELATION• INFO• NONE

Attribute	Possible Values
origin	<ul style="list-style-type: none"> • MANAGEMENTSOFTWARE • MANUALLYCREATED • SYMPTOM • REMOTELYGENERATED • SNMPTRAP • SYSLOG • OTHER
family	<ul style="list-style-type: none"> • com.hp.nms.incident.family.Address • com.hp.nms.incident.family.Interface • com.hp.nms.incident.family.Node • com.hp.nms.incident.family.OSPF • com.hp.nms.incident.family.HSRP • com.hp.nms.incident.family.AggregatePort • com.hp.nms.incident.family.Board • com.hp.nms.incident.family.Connection • com.hp.nms.incident.family.Correlation
category	<ul style="list-style-type: none"> • com.hp.nms.incident.category.Fault • com.hp.nms.incident.category.Status • com.hp.nms.incident.category.Config • com.hp.nms.incident.category.Accounting • com.hp.nms.incident.category.Performance • com.hp.nms.incident.category.Security • com.hp.nms.incident.category.Alert
severity	<ul style="list-style-type: none"> • NORMAL • WARNING • MINOR • MAJOR • CRITICAL

- 5 Repeat [step 4](#) until all filter entries are defined.
- 6 Click **Submit** at the bottom of the form.

Example Incident Filters

Forward NodeDown Incidents from NNMi to HPOM

```
name=NodeDown
```

Forward NodeDown and InterfaceDown Incidents from NNMi to HPOM

```
name=NodeDown
name=InterfaceDown
```

Forward CiscoLinkDown Incidents from NNMi to HPOM

```
name=CiscoLinkDown
```

Forward NNMi Management Events with Severity of MAJOR or MINOR

```
origin=MANAGEMENTSOFTWARE  
severity=MAJOR  
severity=MINOR
```

Forward NNMi Incidents with Severity of at least MINOR and nature of ROOTCAUSE or SERVICEIMPACT

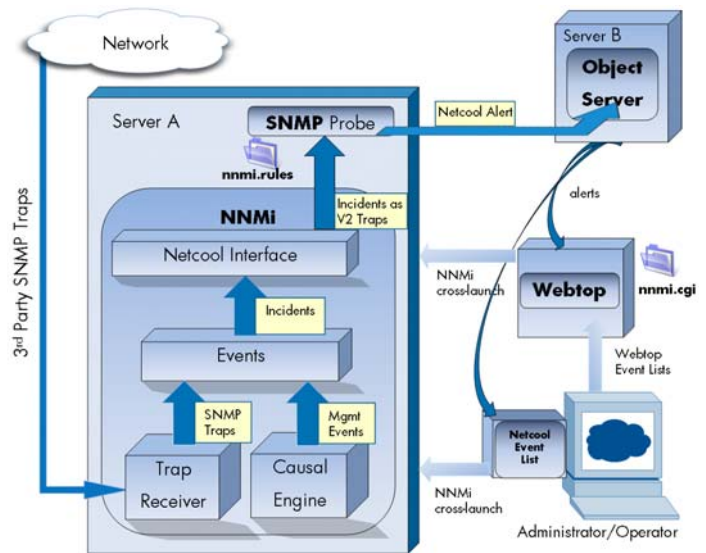
```
severity>=MINOR  
nature=ROOTCAUSE  
nature=SERVICEIMPACT
```

Incident Filter Limitations

Because all filter entries combine to create one incident filter for the NNMi management server, the following limitations apply:

- The stated severity applies to all incidents. For example, to forward NodeDown incidents with a severity of MINOR or higher and InterfaceDown incidents with a severity of MAJOR, set the filter severity to `>=MINOR` and use HPOM logic to filter out the unwanted InterfaceDown messages.
- The incident filter does not provide a mechanism for limiting incident forwarding to specific source nodes. The HPOM managed node (or external node) configuration limits the forwarded incidents that HPOM accepts.

HP NNMi Integration Module for Netcool Software



IBM Tivoli Netcool/OMNIbus consolidates events from a wide range of sources into a single view.

This chapter contains the following topics:

- [HP NNMi Integration Module for Netcool Software](#)
- [Enabling the HP NNMi Integration Module for Netcool Software](#)
- [Using the HP NNMi Integration Module for Netcool Software](#)
- [Changing the HP NNMi Integration Module for Netcool Software](#)
- [Disabling the HP NNMi Integration Module for Netcool Software](#)
- [Troubleshooting the HP NNMi Integration Module for Netcool Software](#)
- [HP NNMi Integration Module for Netcool Software Destination Form Reference](#)

HP NNMi Integration Module for Netcool Software

The HP NNMi Integration Module for Netcool Software forwards NNMi management events as SNMPv2c traps to a Netcool/OMNIbus SNMP Probe on the NNMi management server. The probe filters the NNMi traps and forwards them to the Netcool/OMNIbus server.

While the integration can also forward the SNMP traps that NNMi receives from managed devices to the probe, it is recommended that you instead use the NNMi SNMP trap forwarding mechanism. For more information, see the `hp-nnmi-nbi.mib` file.

The integration does not forward events generated by NNM 6.x or 7.x management stations to the probe.

The integration provides menu items that extend the Netcool event viewers for launching NNMi forms and views in the context of a selected event.

The NNMi Integration Module for Netcool Software is a specific implementation of the NNMi northbound interface, which is described in [NNMi Northbound Interface](#) on page 455.

The NNMi Integration Module for Netcool Software consists of the following components:

- nnmi-northbound integration module
- Configuration files for converting NNMi traps to Netcool/OMNIBus events and creating new menus in the Netcool/Webtop event lists and the Netcool/OMNIBus Event List

Value

The NNMi Integration Module for Netcool Software adds network-level fault and performance information to Netcool/OMNIBus, so that Netcool/OMNIBus users can detect and investigate potential network problems.

The primary features of the integration are as follows:

- Automatic management event forwarding from NNMi to Netcool/OMNIBus. Forwarded management events appear in the Netcool/Webtop event lists and the Netcool/OMNIBus Event List.
- Access to the NNMi console from Netcool/Webtop and Netcool/OMNIBus.
 - Netcool users can open an NNMi form (for example, the Node form) in the context of a selected event and topology object.
 - Netcool users can open an NNMi view (for example, the Layer 2 Neighbor view) in the context of a selected event and node.
 - Netcool users can open the NNMi Incident form in the context of a selected event.

Supported Versions

The information in this chapter applies to the following product versions:

- Netcool/OMNIBus version 7.2.1
- Netcool/OMNIBus SNMP Probe version 7.2.1



The NNMi Integration Module for Netcool Software has been tested with the listed versions of the Netcool products. While the integration should work correctly with other versions of the Netcool products, HP supports the integration only with the listed versions of the Netcool products.

- NNMi version 9.00 or higher, with an NNMi Integration Module for Netcool Software license



As of NNMi 9.00, installing NNMi enables a temporary Instant-On license key for the NNMi Integration Module for Netcool Software. To use the integration after the Instant-On license key expires, obtain and install a permanent license key for the NNMi Integration Module for Netcool Software.

NNMi and Netcool/OMNIBus must be installed on separate computers. The NNMi management server and the Netcool/OMNIBus server computer can be of the same or different operating systems.

The Netcool/OMNIBus SNMP Probe must be installed on the NNMi management server computer.

For the most recent information about supported hardware platforms and operating systems, see the NNMi support matrix and the Netcool/OMNIbus product documentation.

Documentation

This chapter describes how to configure the NNMi Integration Module for Netcool Software to forward NNMi management events to a Netcool/OMNIbus SNMP Probe. It also describes how to use the integration functionality.

For information about Netcool/OMNIbus, see that application's documentation.

Enabling the HP NNMi Integration Module for Netcool Software

The NNMi Integration Module for Netcool Software includes files for configuring the Netcool/OMNIbus SNMP Probe and the Netcool event viewers. Because Netcool is highly configurable, the instructions for the Netcool side of the configuration might not exactly match your Netcool system. It is recommended that an experienced Netcool administrator complete the procedure for enabling the integration.

NNMi Integration Module for Netcool Software Upgraded from NNMi 8.1x

Upgrading to NNMi 9.00 moves the NNMi Integration Module for Netcool Software configuration to the **HP NNMi Integration Module for Netcool Software Destination** form. (To access this form, click **Integration Module Configuration > Netcool**, select a destination, and then click **Edit**.) Verify that the upgraded configuration is accurate.

Table 58 describes the mapping of the NNMi Integration Module for Netcool Software configuration parameters from NNMi 8.13 to NNMi 9.00.

Table 58 NNMi Integration Module for Netcool Software Integration Configuration Parameters from NNMi 8.13 to NNMi 9.00

NNMi 8.13 NNMi Integration Module for Netcool Software Parameter	NNMi 9.00 NNMi Integration Module for Netcool Software Parameter
Use NNMi FQDN Trap Destination Host	Maps to the NNMi FQDN option of the Host field.
Trap Destination Port	Maps to the Port field.
Trap Community String	Maps to the Community String field.
Holding period (minutes)	No direct mapping. The Dampening tab on the incident configuration form defines the holding time per incident, or use <code>nnmsetdampenedinterval.ovpl</code> to set the same holding time for all incident configurations. For more information, see the <code>nnmsetdampenedinterval.ovpl</code> reference page, or the UNIX manpage.
Send 3rd Party Traps	Not selected maps to Management incidents. Selected maps to Both incidents.
NNMi uses https	Maps to the NNMi Console Access field.

Table 58 NNMi Integration Module for Netcool Software Integration Configuration Parameters from NNMi 8.13 to NNMi 9.00

NNMi 8.13 NNMi Integration Module for Netcool Software Parameter	NNMi 9.00 NNMi Integration Module for Netcool Software Parameter
Allow Loopback	Not selected maps to the NNMi FQDN option or the Other option of the Host field. Selected maps to the Use Loopback option of the Host field.
Trap Destination IP Address	Same.
Uptime (seconds)	Same.
Last Sweep Time	Not available. No change to integration behavior.
NNMi URL	Same.
Dampened Stream Enabled All Stream Enabled	Fields not available. No change to integration behavior.
None	Other fields are set to the default behavior for the northbound interface as described in NNMi Northbound Interface Integration Content on page 467.

New NNMi Integration Module for Netcool Software Configuration

To enable the NNMi Integration Module for Netcool Software, follow these steps:

- 1 Gather the information for configuring Netcool:
 - a On any computer, sign in to the NNMi console as an NNMi user with the Administrator role.
 - b In the NNMi console, open the **HP NNMi Integration Module for Netcool Software Configuration Actions** form (**Integration Module Configuration > Netcool**).
 - c Download the rules include file for the Netcool/OMNIBus SNMP Probe by right-clicking the **nnmi.include.rules** link and then saving the file to a known place on the computer.

The `nnmi.include.rules` file defines a rule for interpreting the SNMPv2c traps of the NNMi management events.
 - For information about the contents and format of the traps that NNMi sends to the probe, see the `hp-nnmi-nbi.mib` file.
 - For information about customizing the `nnmi.include.rules` file, see the Netcool/OMNIBus documentation.
 - d *Optional.* Download the information for configuring the Netcool/Webtop event lists to launch NNMi views. Do both of the following:
 - Right-click the **nnmi_launch.cgi** link, and then save the file to a known place on the computer.
 - Right-click the **nnmi_launch_cfg.txt** link, and then save the file to a known place on the computer.

- e *Optional.* Download the information for configuring the Netcool/OMNIbus Event List to launch NNMI views. Do one of the following:
 - *Windows* Netcool/OMNIbus server:
Right-click the **nnmi_confpack.zip** link, and then save the file to a known place on the computer.
 - *UNIX* Netcool/OMNIbus server:
Right-click the **nnmi_confpack.gz** link, and then save the file to a known place on the computer.
- 2 Install the Netcool/OMNIbus SNMP Probe on the NNMI management server.
 - a Configure the probe to receive SNMP traps on an available UDP port.
 - Note this port number for configuring the integration in NNMI.
 - Verify that the probe port is different from the port on which NNMI receives SNMP traps, which is configured on the **Communication Configuration** form in the NNMI console.
 - b Copy the `nnmi.include.rules` file from [step 1c](#) to the NNMI management server.
 - c Back up the master rules file, and then open the file in any text editor.
 - d Within the Netcool enterprise trap switch block, add an `include` directive for the `nnmi.include.rules` file, and then save the master rules file.
 - e Restart the probe, and then examine the probe log file to verify that there were no problems reloading the rules file.

For more information about installing and configuring the probe, see the probe documentation.

- 3 Configure NNMI incident forwarding:
 - a On any computer, sign in to the NNMI console as an NNMI user with the Administrator role.
 - b In the NNMI console, open the **HP NNMI Integration Module for Netcool Software Configuration Actions** form (**Integration Module Configuration > Netcool**).
 - c Click **Enable/Disable** NNMI Integration Module for Netcool Software, and then click **New**.

(If you have selected an available destination, click **Reset** to make the **New** button available.)
 - d On the **HP NNMI Integration Module for Netcool Software Destination** form, select the **Enabled** check box to make the remaining fields on the form available.
 - e Enter the information for connecting to the Netcool/OMNIbus SNMP Probe.
For information about these fields, see [Netcool/OMNIbus SNMP Probe Connection](#) on page 534.
 - f Specify the sending options.
For information about these fields, see [Integration Content](#) on page 535.
 - g Click **Submit** at the bottom of the form.

A new window displays a status message. If the message indicates a problem with the settings, click **Return**, and then adjust the values as suggested by the text of the error message.

- 4 *Optional.* Configure the Netcool/Webtop event lists to launch NNMi views.
 - a Copy the `nnmi_launch.cgi` file from [step 1d](#) to the `cgi-bin` directory on the Netcool/Webtop server.
 - b Follow the instructions in the `nnmi_launch_cfg.txt` file from [step 1d](#) to prepare the CGI file and to configure the Netcool/Webtop menus.
- 5 *Optional.* Configure the Netcool/OMNIbus Event List to launch NNMi views.
 - a Copy the `nnmi_confpack.*` archive file from [step 1e](#) to the computer where the instance of the Netcool/OMNIbus ObjectServer is running.
 - b Unpack the `nnmi_confpack.*` archive file to a temporary location.
 - c From the temporary location, run the following command:
 - *Windows* Netcool/OMNIbus server:


```
%OMNIBUSHOME%\bin\nco_confpack -import \  
-package nnmi.confpack \  
-user <objectserver_administrator_username> \  
-server <objectserver_name>
```
 - *UNIX* Netcool/OMNIbus server:


```
$OMNIBUSHOME/bin/nco_confpack -import \  
-package nnmi.confpack \  
-user <objectserver_administrator_username> \  
-server <objectserver_name>
```
 - d *UNIX only:* Verify that `$OMNIBROWSER` is set to the location of the Mozilla Firefox browser.

Using the HP NNMi Integration Module for Netcool Software

When the NNMi Integration Module for Netcool Software is enabled, NNMi sends SNMPv2c traps to the Netcool/OMNIbus SNMP Probe. View the content forwarded from NNMi in the Netcool/Webtop event lists and the Netcool/OMNIbus Event List.

For information about the types of traps the integration module can forward to the probe, see [Using the NNMi Northbound Interface](#) on page 459. For information about the contents and format of these traps, see the `hp-nnmi-nbi.mib` file. For a comparison of trap forwarding mechanisms, see [Trap and Incident Forwarding](#) on page 87.

NNMi sends only one copy of each management event trap (or received SNMP trap) to the Netcool/OMNIbus SNMP Probe. NNMi does not queue traps. If the probe is unavailable when NNMi forwards a trap, that trap is lost.

The integration module provides links to the NNMi console from the Netcool event viewers. Enter your NNMi user credentials to see the NNMi console views.

In [Enabling the HP NNMi Integration Module for Netcool Software](#) on page 527, [step 4](#) and [step 5](#) add the following menu items to the Netcool event viewers:

- **Source Object**—Opens the NNMi form for the object in the event selected in Netcool/OMNIbus.
- **Node**—Opens the NNMi Node form for the node in the event selected in Netcool/OMNIbus.

- **L2 Neighbors**—Opens the NNMi Layer 2 Neighbor View for the node in the event selected in Netcool/OMNIBus.
- **L3 Neighbors**—Opens the NNMi Layer 3 Neighbor View for the node in the event selected in Netcool/OMNIBus.
- **Incident Details**—Opens the NNMi Incident form for the event selected in Netcool/OMNIBus.



On UNIX Netcool/OMNIBus servers:

- Mozilla Firefox must be the default web browser to support the launching of NNMi views from the Netcool/OMNIBus Event List.
- The `$OMNIBROWSER` environment variable must be set to the location of the Mozilla Firefox browser.

Changing the HP NNMi Integration Module for Netcool Software

To change the NNMi Integration Module for Netcool Software configuration parameters, follow these steps:

- 1 In the NNMi console, open the **HP NNMi Integration Module for Netcool Software Configuration Actions** form (**Integration Module Configuration > Netcool**).
- 2 Click **Enable/Disable** NNMi Integration Module for Netcool Software.
- 3 Select a destination, and then click **Edit**.
- 4 Modify the values as appropriate.

For information about the fields on this form, see [HP NNMi Integration Module for Netcool Software Destination Form Reference](#) on page 534.

- 5 Verify that the **Enable** check box at the top of the form is selected, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

Disabling the HP NNMi Integration Module for Netcool Software

No SNMP trap queuing occurs while a destination is disabled.

To discontinue the forwarding of NNMi management events to the Netcool/OMNIBus SNMP Probe, follow these steps:

- 1 In the NNMi console, open the **HP NNMi Integration Module for Netcool Software Configuration Actions** form (**Integration Module Configuration > Netcool**).
- 2 Click **Enable/Disable** NNMi Integration Module for Netcool Software.
- 3 Select a destination, and then click **Edit**.

Alternatively, click **Delete** to entirely remove the configuration for the selected destination.

- 4 On the **HP NNMi Integration Module for Netcool Software Destination** form, clear the **Enable** check box at the top of the form, and then click **Submit** at the bottom of the form.

The changes take effect immediately.

- 5 To conserve system resources, shut down the Netcool/OMNIBus SNMP Probe while the destination is disabled.

To permanently disable the integration, also do the following:

- Uninstall the Netcool/OMNIBus SNMP Probe as described in the probe documentation.
- Remove the NNMi menu items from the Netcool/Webtop and Netcool/OMNIBus Event List configurations.

Troubleshooting the HP NNMi Integration Module for Netcool Software

Netcool/OMNIBus Does Not Receive Any Forwarded NNMi Management Events

If the Netcool event viewer does not contain any traps from NNMi, follow these steps:

- 1 Verify that the Netcool/OMNIBus SNMP Probe is receiving traps:
 - a Verify that the probe can send messages to the Netcool/OMNIBus server.
 - b Verify that the probe master rules file includes or contains the content of the `nnmi.include.rules` file.
 - c Verify the syntax of the master rules file.
 - d Examine the probe log file to verify that there were no problems loading the rules file.
 - e Examine the probe log file to determine whether the NNMi traps arrive at the probe.
 - f Examine the probe log file to determine whether the probe processes or drops the incoming traps.

For information about troubleshooting the probe, see the Netcool/OMNIBus documentation.

- 2 Verify that NNMi is forwarding management events to the Netcool/OMNIBus SNMP Probe.

For information, see [Troubleshooting the NNMi Northbound Interface](#) on page 463.

Netcool/OMNIBus Does Not Receive Some Forwarded NNMi Management Events

If one or more NNMi management event traps do not appear in the Netcool event viewer, follow these steps:

- 1 Verify that the Netcool/OMNIBus SNMP Probe master rules file includes or contains the content of the `nnmi.include.rules` file.
- 2 Verify that Netcool/OMNIBus is running.

If the Netcool/OMNIBus server shuts down, the Netcool/OMNIBus SNMP Probe queues received traps. The probe forwards the queued traps when the Netcool/OMNIBus server becomes available.

NNMi relies on the probe to queue and forward traps. If the probe shuts down, the forwarded traps are lost.

- 3 Verify that the NNMi processes are running.

Error When Launching an NNMi Form for a Layer 2 Connection

If the source object in an NNMi management event is a layer 2 connection, NNMi users with a role other than Administrator cannot open the NNMi form directly from the **Source Object** menu item in the Netcool event viewer. Instead, in the Netcool event viewer, use the **L2 Neighbors** menu item to connect to NNMi, and then double-click the connection in the Layer 2 Neighbor View.

HP NNMi Integration Module for Netcool Software Destination Form Reference

The **HP NNMi Integration Module for Netcool Software Destination** form contains the parameters for configuring communications between NNMi and a Netcool/OMNIBus SNMP Probe. When a valid NNMi Integration Module for Netcool Software license has been installed on the NNMi management server, this form is available from the **Integration Module Configuration** workspace. (On the **HP NNMi Integration Module for Netcool Software Configuration Actions** form, click **Enable/Disable** NNMi Integration Module for Netcool Software. Click **New**, or select a destination, and then click **Edit**.)



Only NNMi users with the Administrator role can access the **HP NNMi Integration Module for Netcool Software Destination** form.

The **HP NNMi Integration Module for Netcool Software Destination** form collects information for the following areas:

- [Netcool/OMNIBus SNMP Probe Connection](#) on page 534
- [Integration Content](#) on page 535
- [Destination Status Information](#) on page 538

To apply changes to the integration configuration, update the values on the **HP NNMi Integration Module for Netcool Software Destination** form, and then click **Submit**.

Netcool/OMNIBus SNMP Probe Connection

[Table 59](#) lists the parameters for configuring the connection to the Netcool/OMNIBus SNMP Probe.

Table 59 Netcool/OMNIBus SNMP Probe Connection Information

Field	Description
Host	<p>The fully-qualified domain name (preferred) or the IP address of the NNMi management server, which is the system on which the Netcool/OMNIBus SNMP Probe receives SNMP traps from NNMi.</p> <p>The integration supports the following methods for identifying the probe host:</p> <ul style="list-style-type: none">• NNMi FQDN NNMi manages the connection to the probe on the NNMi management server and the Host field becomes read-only. This is the default and recommended configuration.• Use Loopback NNMi manages the connection to the probe on the NNMi management server and the Host field becomes read-only.• Other Do not use this option. <p>NOTE: If the NNMi management server participates in NNMi application failover, see Application Failover and the NNMi Northbound Interface on page 465 for information about the impact of application failover on the integration.</p>

Table 59 Netcool/OMNIBus SNMP Probe Connection Information (cont'd)

Field	Description
Port	<p>The UDP port where the Netcool/OMNIBus SNMP Probe receives SNMP traps. Enter the port number specific to the probe.</p> <p>To determine the port, examine the probe <code>mttrapd.properties</code> file on the NNMi management server.</p> <p>NOTE: This port number must be different from the port on which NNMi receives SNMP traps, as set in the SNMP Port field on the Communication Configuration form in the NNMi console.</p>
Community String	<p>A read-only community string for the Netcool/OMNIBus SNMP Probe to receive traps.</p> <p>If the probe configuration requires a specific community string in the received SNMP traps, enter that value.</p> <p>If the probe configuration does not require a specific community string, use the default value, which is <code>public</code>.</p>

Integration Content

Table 60 lists the parameters for configuring which content the NNMi Integration Module for Netcool Software sends to the Netcool/OMNIBus SNMP Probe.

Table 60 NNMi Integration Module for Netcool Software Content Configuration Information

Field	Description
Incidents	<p>The incident forwarding specification.</p> <ul style="list-style-type: none"> • Management NNMi forwards only NNMi-generated management events to the Netcool/OMNIBus SNMP Probe. This is the default configuration. • SNMP 3rd Party Trap NNMi forwards only SNMP traps that NNMi receives from managed devices to the probe. • Both NNMi forwards to the probe both NNMi-generated management events and SNMP traps that NNMi receives from managed devices. <p>NNMi begins forwarding incidents as soon as you enable the destination. For more information, see Incident Forwarding on page 459.</p>

Table 60 NNMi Integration Module for Netcool Software Content Configuration Information

Field	Description
Lifecycle State Changes	<p>The incident change notification specification.</p> <ul style="list-style-type: none">• Enhanced Closed NNMi sends an incident closed trap to the Netcool/OMNIBus SNMP Probe for each incident that changes to the CLOSED lifecycle state. This is the default configuration.• State Changed NNMi sends an incident lifecycle state changed trap to the probe for each incident that changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state.• Both NNMi sends an incident closed trap to the probe for each incident that changes to the CLOSED lifecycle state. Additionally, the integration sends an incident lifecycle state changed trap to the probe for each incident that changes to the IN PROGRESS, COMPLETED, or CLOSED lifecycle state. NOTE: In this case, each time an incident changes to the CLOSED lifecycle state, the integration sends two notification traps: an incident closed trap and an incident lifecycle state changed trap. <p>For more information, see Incident Lifecycle State Change Notifications on page 460.</p>
Correlations	<p>The incident correlation notification specification.</p> <ul style="list-style-type: none">• None NNMi does not notify the Netcool/OMNIBus SNMP Probe of incident correlations resulting from NNMi causal analysis. This is the default configuration.• Single NNMi sends a trap for each parent-child incident correlation relationship resulting from NNMi causal analysis.• Group NNMi sends one trap per correlation that lists all child incidents correlated to a parent incident. <p>For more information, see Incident Correlation Notifications on page 461.</p>
Deletions	<p>The incident deletion specification.</p> <ul style="list-style-type: none">• Don't Send NNMi does not notify the Netcool/OMNIBus SNMP Probe when incidents are deleted in NNMi. This is the default configuration.• Send NNMi sends a deletion trap to the probe for each incident that is deleted in NNMi. <p>For more information, see Incident Deletion Notifications on page 462.</p>

Table 60 NNMi Integration Module for Netcool Software Content Configuration Information

Field	Description
NNMi Console Access	<p>The connection protocol specification in the URL for browsing to the NNMi console from the Netcool event viewer. The traps that NNMi sends to the Netcool/OMNIBus SNMP Probe include the NNMi URL in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2).</p> <p>The configuration page defaults to the setting that matches the NNMi configuration. If the NNMi console is configured to accept both HTTP and HTTPS connections, you can change the HTTP connection protocol specification in the NNMi URL. For example, if all Netcool users are on the intranet, you can set NNMi console access from the Netcool event viewer to be over HTTP. To change the protocol for connecting to the NNMi console from the Netcool event viewer, select the HTTP option or the HTTPS option as appropriate.</p>
Incident Filters	<p>A list of object identifiers (OIDs) on which the integration filters the events sent to the Netcool/OMNIBus SNMP Probe. Each filter entry can be a valid numeric OID (for example, .1.3.6.1.6.3.1.1.5.4.1.3.6.1.4.1.9) or OID prefix (for example, .1.3.6.1.6.3.1.1.5.*).</p> <p>Select one of the following options:</p> <ul style="list-style-type: none">• None NNMi sends all events to the probe. This is the default configuration.• Include NNMi sends only the specific events that match the OIDs identified in the filter.• Exclude NNMi sends all events except for the specific events that match the OIDs identified in the filter. <p>Specify the incident filter:</p> <ul style="list-style-type: none">• To add a filter entry, enter the text in the lower text box, and then click Add.• To delete a filter entry, select that entry from the list in the upper box, and then click Remove. <p>For more information, see Event Forwarding Filter on page 462.</p>

Destination Status Information

Table 61 lists the read-only status information for the NNMi Integration Module for Netcool Software destination. This information is useful for verifying that the integration is working correctly.

Table 61 NNMi Integration Module for Netcool Software Status Information

Field	Description
Trap Destination IP Address	The IP address to which the Netcool/OMNIBus SNMP Probe destination host name resolves. This value is unique to this probe destination.
Uptime (seconds)	The time (in seconds) since the northbound component was last started. The traps that NNMi sends to the Netcool/OMNIBus SNMP Probe include this value in the sysUptime field (1.3.6.1.2.1.1.3.0). This value is the same for all integrations that use the NNMi northbound interface. To see the latest value, either refresh or close and re-open the form.
NNMi URL	The URL for connecting to the NNMi console. The traps that NNMi sends to the Netcool/OMNIBus SNMP Probe include this value in the NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2). This value is unique to this northbound destination.

Additional Information

This section contains the following appendices:

- [NNMi Environment Variables](#)
- [NNMi 9.00 and Well-Known Ports](#)
- [Suggested Configuration Changes](#)

NNMi Environment Variables

HP Network Node Manager i Software provides many environment variables that are available for your use in navigating the file system and writing scripts.

This appendix contains the following topics:

- [Environment Variables Used in This Document](#)
- [Other Available Environment Variables](#)
- [Windows Paths and Environment Variables from NNMi 8.00](#)

Environment Variables Used in This Document

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

- *Windows Server 2003:*
 - %NnmInstallDir%: <drive>\Program Files\HP\HP BTO Software
 - %NnmDataDir%: <drive>\Documents and Settings\All Users\Application Data\HP\HP BTO Software



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



If you first installed NNMi 8.00, your system uses different values for these environment variables, as described in [Windows Paths and Environment Variables from NNMi 8.00](#) on page 545.

- *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](#) on page 542.

Other Available Environment Variables

NNMi administrators access some NNMi file locations regularly. NNMi provides a script that sets up many environment variables for navigating to commonly accessed locations.



If you first installed NNMi 8.00 on the Windows operating system, see [Windows Paths and Environment Variables from NNMi 8.00](#) on page 545.

To set up the extended list of NNMi environment variables, use a command similar to the following examples:

- Windows: "C:\Program Files\HP\HP BTO Software\bin\nnm.envvars.bat"
- UNIX: . /opt/OV/bin/nnm.envvars.sh

After you run the command for your operating system, you can use the NNMi environment variables shown in [Table 62](#) (Windows) or [Table 63](#) (UNIX) to get to commonly used NNMi file locations.

Table 62 Environment Variable Default Locations for the Windows Operating System

Variable	Windows (example)
%NNM_BIN%	C:\Program Files\HP\HP BTO Software\bin
%NNM_CONF%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\conf • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\conf
%NNM_DATA%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\ • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\
%NNM_DB%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\databases • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\databases
%NNM_JAVA%	C:\Program Files\HP\HP BTO Software\nonOV\jdk\b\bin\java.exe
%NNM_JAVA_DIR%	C:\Program Files\HP\HP BTO Software\java
%NNM_JAVA_PATH_SEP%	;
%NNM_JBOSS%	C:\Program Files\HP\HP BTO Software\nonOV\jboss\nms
%NNM_JBOSS_DEPLOY%	C:\Program Files\HP\HP BTO Software\nonOV\jboss\nms\server\nms\deploy

Table 62 Environment Variable Default Locations for the Windows Operating System (cont'd)

Variable	Windows (example)
%NNM_JBOSS_LOG%	C:\Program Files\HP\HP BTO Software\nonOV\jboss\nms\server\nms\log
%NNM_JBOSS_ROOT%	C:\Program Files\HP\HP BTO Software\nonOV\jboss\nms
%NNM_JBOSS_SERVERCONF%	C:\Program Files\HP\HP BTO Software\nonOV\jboss\nms\server\nms
%NNM_JRE%	C:\Program Files\HP\HP BTO Software\nonOV\jdk\b
%NNM_LOG%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\log • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\log
%NNM_LRF%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\shared\nnm\lrf • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\shared\nnm\lrf
%NNM_PRIV_LOG%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\log • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\log
%NNM_PROPS%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\shared\nnm\conf\props • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\shared\nnm\conf\props
%NNM_SHARED_CONF%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\shared\nnm\conf • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\shared\nnm\conf
%NNM_SHARE_LOG%	<ul style="list-style-type: none"> • <i>Windows Server 2008:</i> C:\ProgramData\HP\HP BTO Software\log • <i>Windows Server 2003:</i> C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\log

Table 62 Environment Variable Default Locations for the Windows Operating System (cont'd)

Variable	Windows (example)
%NNM_SNMP_MIBS%	C:\Program Files\HP\HP BTO Software\misc\nnm\snmp_mibs
%NNM_SUPPORT%	C:\Program Files\HP\HP BTO Software\support
%NNM_TMP%	<ul style="list-style-type: none"> Windows Server 2008: C:\ProgramData\HP\HP BTO Software\tmp Windows Server 2003: C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\tmp
%NNM_USER_SNMP_MIBS%	<ul style="list-style-type: none"> Windows Server 2008: C:\ProgramData\HP\HP BTO Software\shared\nnm\user-snmplib Windows Server 2003: C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\shared\nnm\user-snmplib
%NNM_WWW%	C:\Program Files\HP\HP BTO Software\www

Table 63 Environment Variable Default Locations for UNIX Operating Systems

Variable	HP-UX
\$NNM_BIN	/opt/OV/bin
\$NNM_CONF	/var/opt/OV/conf
\$NNM_DATA	/var/opt/OV
\$NNM_DB	/var/opt/OV/databases
\$NNM_JAVA	/opt/OV/nonOV/jdk/b/bin/java
\$NNM_JAVA_DIR	/opt/OV/java
\$NNM_JAVA_PATH_SEP	:
\$NNM_JBOSS	/opt/OV/nonOV/jboss/nms
\$NNM_JBOSS_DEPLOY	/opt/OV/nonOV/jboss/nms/server/nms/deploy
\$NNM_JBOSS_LOG	/opt/OV/nonOV/jboss/nms/server/nms/log
\$NNM_JBOSS_ROOT	/opt/OV/nonOV/jboss/nms
\$NNM_JBOSS_SERVERCONF	/opt/OV/nonOV/jboss/nms/server/nms
\$NNM_JRE	/opt/OV/nonOV/jdk/b
\$NNM_LOG	/var/opt/OV/log
\$NNM_LRF	/var/opt/OV/shared/nm/lrf
\$NNM_PRIV_LOG	/var/opt/OV/log

Table 63 Environment Variable Default Locations for UNIX Operating Systems (cont'd)

Variable	HP-UX
\$NNM_PROPS	/var/opt/OV/shared/nnm/conf/props
\$NNM_SHARED_CONF	/var/opt/OV/shared/nnm/conf
\$NNM_SHARE_LOG	/var/opt/OV/log
\$NNM_SNMP_MIBS	/opt/OV/misc/nnm/snmp_mibs
\$NNM_SUPPORT	/opt/OV/support
\$NNM_TMP	/var/opt/OV/tmp
\$NNM_USER_SNMP_MIBS	/var/opt/OV/shared/nnm/user-snmp-mibs
\$NNM_WWW	/opt/OV/www

Windows Paths and Environment Variables from NNMi 8.00

On the Windows operating system, the default paths to NNMi files are different for versions 8.00 and 8.01 (or higher). When you upgrade to a newer version of NNMi on top of NNMi 8.00, the newer version of NNMi continues to use the paths that were defined for NNMi 8.00. In this case, the locations of the environment variables created during NNMi installation are as follows:

- %NnmInstallDir%: <drive>\Program Files (x86)\HP OpenView
- %NnmDataDir%: <drive>\Program Files (x86)\HP OpenView\data

To set up the extended list of NNMi environment variables, use a command similar to the following example:

```
"C:\Program Files (x86)\HP OpenView\bin\nnm.envvars.bat"
```

After you run the command, you can use the NNMi environment variables shown in [Table 64](#) to get to commonly used NNMi file locations.



For the upgrade scenario *only*, the information in [Table 64](#) supersedes that in [Table 62](#) on page 542.

Table 64 NNMi 8.00 Windows Operating System Environment Variable Default Locations

Variable	Windows (example)
%NNM_BIN%	C:\Program Files (x86)\HP OpenView\bin
%NNM_CONF%	C:\Program Files (x86)\HP OpenView\data\conf
%NNM_DATA%	C:\Program Files (x86)\HP OpenView\data
%NNM_DB%	C:\Program Files (x86)\HP OpenView\data\databases
%NNM_JAVA%	C:\Program Files (x86)\HP OpenView\nonOV\jdk\b\bin\java.exe
%NNM_JAVA_DIR%	C:\Program Files (x86)\HP OpenView\java

Table 64 NNMi 8.00 Windows Operating System Environment Variable Default Locations

Variable	Windows (example)
%NNM_JAVA_PATH_SEP%	;
%NNM_JBOSS%	C:\Program Files (x86)\HP OpenView\nonOV\jboss\nms
%NNM_JBOSS_DEPLOY%	C:\Program Files (x86)\HP OpenView\nonOV\jboss\nms\server\nms\deploy
%NNM_JBOSS_LOG%	C:\Program Files (x86)\HP OpenView\nonOV\jboss\nms\server\nms\log
%NNM_JBOSS_ROOT%	C:\Program Files (x86)\HP OpenView\nonOV\jboss\nms
%NNM_JBOSS_SERVERCONF%	C:\Program Files (x86)\HP OpenView\nonOV\jboss\nms\server\nms
%NNM_JRE%	C:\Program Files (x86)\HP OpenView\nonOV\jdk\b
%NNM_LOG%	C:\Program Files (x86)\HP OpenView\data\log
%NNM_LRF%	C:\Program Files (x86)\HP OpenView\data\shared\nnm\lrf
%NNM_PRIV_LOG%	C:\Program Files (x86)\HP OpenView\data\log
%NNM_PROPS%	C:\Program Files (x86)\HP OpenView\data\shared\nnm\conf\props
%NNM_SHARE_LOG%	C:\Program Files (x86)\HP OpenView\data\log
%NNM_SHARED_CONF%	C:\Program Files (x86)\HP OpenView\data\shared\nnm\conf
%NNM_SNMP_MIBS%	C:\Program Files (x86)\HP OpenView\misc\nnm\snmp_mibs
%NNM_SUPPORT%	C:\Program Files (x86)\HP OpenView\support
%NNM_TMP%	C:\Program Files (x86)\HP OpenView\data\tmp
%NNM_USER_SNMP_MIBS%	C:\Program Files (x86)\HP OpenView\data\shared\nnm\user-snm-mibs
%NNM_WWW%	C:\Program Files (x86)\HP OpenView\www

NNMi 9.00 and Well-Known Ports

Table 65 shows the ports NNMi uses on the management server. NNMi listens on these ports. If port conflicts occur, you can change most of these port numbers as shown in the *Change Configuration* column. See the *nnm.port.4* reference page, or the UNIX manpage, for more information.

▶ For application failover to function correctly, the active and standby NNMi management servers must have unrestricted network access to each other.

Table 65 Ports Used on the NNMi Management Server

Port	Type	Name	Purpose	Change Configuration
80	TCP	jboss.http.port	Default HTTP port - used for Web UI & Web Services	Modify the <code>nms-local.properties</code> file You can also change this during installation
162	UDP	trapPort	SNMP trap port	Modify using the <code>nnmtrapconfig.ovpl</code> Perl script. See the <i>nnmtrapconfig.ovpl</i> reference page, or the UNIX manpage, for more information.
443	TCP	jboss.https.port	Default secure HTTPS port (SSL) - used for Web UI & Web Services	Modify the <code>nms-local.properties</code> file
1098	TCP	jboss.rmi.port	Default port for RMI naming service	Modify the <code>nms-local.properties</code> file
1099	TCP	jboss.jnp.port	Default bootstrap JNP service port (JNDI provider)	Modify the <code>nms-local.properties</code> file
3873	TCP	jboss.ejb3.port	Default EJB3 remoting connector port	Modify the <code>nms-local.properties</code> file
4444	TCP	jboss.jrmp.port	Default RMI object port (JRMP invoker)	Modify the <code>nms-local.properties</code> file
4445	TCP	jboss.pooled.port	Default RMI pooled invoker port	Modify the <code>nms-local.properties</code> file

Table 65 Ports Used on the NNMi Management Server

Port	Type	Name	Purpose	Change Configuration
4446	TCP	jboss.socket.port	Default RMI remotng server connector port	Modify the nms-local.properties file
4457	TCP	jboss.bisocket.port	Default messaging bi-socket connector	Modify the nms-local.properties file
4458	TCP	jboss.jmsControl.port	Default JMS control port; used for global network management communication	Modify the nms-local.properties file
4459	TCP	jboss.sslbisocket.port	Default messaging bi-socket connector; used for secure global network management communication	Modify the nms-local.properties file
4460	TCP	jboss.ssljmsControl.port	Default JMS control port; used for secure global network management communication	Modify the nms-local.properties file
5432	TCP		Postgres port	Not configurable
7800-7810	TCP		JGroups multicast ports for multi-subnet application failover	Modify the nms-cluster.properties file
8083	TCP	jboss.ws.port	Default jboss Web Service port	Modify the nms-local.properties file
8886	TCP	OVSPMD_MGMT	NNMi ovspmd (process manager) management port	Modify the /etc/services file
8887	TCP	OVSPMD_REQ	NNMi ovsmppd (process manager) request port	Modify the /etc/services file
45588	UDP	jgroups.udp.mcast_port	JGroups multicast port for LAN application failover	Modify the nms-cluster.properties file

Table 66 shows some of the ports NNMi uses to communicate with other systems. If a firewall separates NNMi from these systems, you need to open many of these ports in the firewall. The actual set of ports depends on the set of integrations you configured to use with NNMi and how you configured those integrations. If column 4 indicates *Client*, NNMi connects or sends to this port; if column 4 indicates *Server*, NNMi listens on this port.

Table 66 Ports Used for Communication Between the NNMi Management Server and Other Systems

Port	Type	Purpose	Client/Server
80	TCP	Default HTTP port for NNMi; used for Web UI and Web Services	Server
80	TCP	Default HTTP port for NNMi connecting to other applications. The actual port depends on NNMi configuration.	Client
161	UDP	SNMP request port	Client
162	UDP	SNMP trap port - traps received by NNMi	Server
162	UDP	SNMP trap port; Trap Forwarding, Northbound Interface, or NetCool integrations	Client
389	TCP	Default LDAP port	Client
395	UDP	nGenius Probe SNMP trap port	Client
443	TCP	Default secure HTTPS port for NNMi connecting to other applications; the actual port depends on NNMi configuration. Default HTTPS port for HP OM on Windows	Client
443	TCP	Default secure HTTPS port; used for Web UI and Web Services	Server
636	TCP	Default secure LDAP port (SSL)	Client
1741	TCP	Default CiscoWorks LMS web services port	Client
4457	TCP	Default messaging bi-socket connector used for global network management communication. The connection is from the global manager to the regional manager.	Client/Server
4458	TCP	Default JMS control port used for global network management communication. The connection is from the global manager to the regional manager.	Client/Server
4459	TCP	Default messaging bi-socket connector used for secure global network management communication. The connection is from the global manager to the regional manager.	Client/Server
4460	TCP	Default JMS control port used for secure global network management communication. The connection is from the global manager to the regional manager.	Client/Server
7800-7810	TCP	JGroups Multicast ports for multi-subnet application failover	Client and Server
8004	TCP	Default HTTP port for NNMi if another web server already has port 80. Used for Web UI and Web Services. Verify the actual HTTP port for your NNMi management server.	Server

Table 66 Ports Used for Communication Between the NNMi Management Server and Other Systems

Port	Type	Purpose	Client/Server
8080	TCP	Default HTTP port for connecting to NA if installed on the same system as NNMi. Default HTTPS port for HP UCMDB web services	Client
8443 or 8444	TCP	Default HTTP port for connecting to HP OM for UNIX	Client
9300	TCP	Default HTTP port for connecting to NNM iSPI for Performance	Client
45588	UDP	JGroups Multicast port for LAN application failover	Client and Server
50000	TCP	Default HTTPS port for connecting to SIM	Client

▶ If you configure NNMi to use ICMP fault polling or ping sweep for discovery, configure the firewall to pass ICMP packets through the firewall.

▶ The Web Services approach for the NNMi-HP OM integration does not work through a firewall, however the NNMi-HP OM integration using the Northbound Interface does work through a firewall.

If you plan to use the global network management feature, [Table 67](#) shows the well-know ports that need to be accessible from a global NNMi management server to a regional NNMi management server. The global network management feature requires these ports to be open for TCP access from the global NNMi management server to the regional NNMi management server. The regional NNMi management server will not open sockets back to the global NNMi management server.

Table 67 Required Accessible Sockets for Global Network Management

Security	Parameter	TCP Port
non-SSL	jboss.http.port	80
	jboss.bisocket.port	4457
	jboss.jmsControl.port	4458
SSL	jboss.https.port	443
	jboss.sslbisocket.port	4459
	jboss.ssljmsControl.port	4460

Suggested Configuration Changes

Some common actions for performance improvements and how to complete them.

Messages and Solutions

Message: The database pool has *number* available connections.

Solution: This is a warning which means that the database pool is being heavily used. If this only appears for a short time then it probably indicates NNMi is under high load. If it appears frequently then it may indicate a performance problem.

Message: The database connection pool is exhausted. NNM should be restarted immediately.

Solution: This is a serious error indicating that the database pool has failed for some reason and NNMi cannot access the database. You need to restart `ovjboss` or contact support to fix the issue.

Message: Detected *number* missing database connections. NNM should be restarted in the near future to correct this issue.

Solution: This is a warning that NNMi detected inconsistencies in the database pool. Restart `ovjboss` at a convenient time to address this issue.

Message: The disk location *location name* only has *number*% free.

Solution: NNMi displays this warning after a disk location used by NNMi gets low on space.

Message: The average system load is *number*.

Solution: NNMi displays this warning after the system load passes a threshold. If it shows occasionally during major activity, you can ignore the message. However, if it shows constantly then you need to investigate if the system is underpowered for the size environment or if there is some other process on the system consuming significant resources.

Message: The system is low on swap space with *number* MB remaining.

Solution: NNMi displays this message after NNMi detects the system is running low on swap space. You should increase the free swap space or reduce the use of swap.

Message: The NNMi process is currently using *number*% of allowed open files.

Solution: This message indicates that NNMi is running out of file handles. It is expected that less than 1,000 files will be open at any one time but certain environments may cause more to be required. If the number of open file handles keeps growing however that may indicate a defect in the product.

Message: The NNMi CPU utilization is at 100%.

Solution: This indicates that the ovjboss process has been consuming 100% CPU on the system for a significant amount of time. This may be expected for brief periods of high load but if this warning appears constantly then the system may be underpowered for the load.

Message: The global manager *hostname* has *number* outstanding messages waiting to be delivered.

Solution: This indicates that a global manager connected to this system is either down or unable to receive messages fast enough. If the global manager is just down for maintenance then it should recover when it comes back online however if this number goes over several hundred thousand then it may need to be manually cleared out (contact support).

Message: The connection to the regional manager *regional name* is down.

Warning message if communication to a regional manager is interrupted. If this appears for a short time then it isn't a cause for concern but if this message appears for an extended period then some troubleshooting may be required. The global manager will not be receiving any state changes from the regional while this condition exists.

Solution: There are *number* out of *number* maximum connections currently open to the embedded database.

Message: This message warns that the embedded database NNMi uses is approaching its limit of open connections. This may indicate the system is under excessive load or that too many SPIs are installed. It is recommended action such as shutting down SPIs is taken before this number reaches the maximum.

Solution: The memory region *region name* is at *number*% usage.

Message: The system has spent *number*% of total uptime in the *collector name* collector.

Solution: Both these messages indicate that ovjboss is low on memory. The memory settings for the specified region should be checked against what is recommended for the environment scale and increased if the system has available memory.

Problems and Solutions

Problem: NNMi does not always interpret and display SNMP data and MIB strings correctly.

Solution: This is caused by NNMi not always knowing which character set to use to interpret this data. The result is that NNMi displays garbled strings from some SNMP traps and other octetstring data, such as `sysDescription`, `sysContact` and other data. The solution is to use the correct character set to interpret this data.

For SNMP traps and other octetstring data that result in garbled text displays due to using improper character sets, do the following:

- a Edit the following file:
 - *Windows:* `%NNM_PROPS%\nms-jboss.properties`
 - *UNIX:* `$NNM_PROPS/nms-jboss.properties`
- b Remove the comment (`#!` characters) from the line that begins as follows:
`#!com.hp.nnm.sourceEncoding=`
- c Set the `com.hp.nnm.sourceEncoding` JVM property to a comma-separated list of source encodings that your environment currently supports using the examples shown in the `nms-jboss.properties` file. These examples show combinations of the `Shift_JIS`, `EUC_JP`, `UTF-8`, and `ISO-8859-1` character sets.
- d Save your changes.
- e From a command prompt, run `ovstop ovjboss`.
- f From a command prompt, run `ovstart ovjboss`.
- g To test your changes, resend the suspect trap to NNMi and make sure the garbled display problem no longer occurs.

If the garbled text involves binary data or data that cannot be interpreted for any reason, do the following to configure NNMi to display the strings in hexadecimal format:

- a Edit the following file:
 - *Windows:* `%NNMDATADIR%\shared\nnm\conf\nnmvbnosrcenc.conf`
 - *UNIX:* `$NNMDATADIR/shared/nnm/conf/nnmvbnosrcenc.conf`
- b Add the trap OID, varbind OID value combinations that NNMi displays in a garbled format. Also add the combinations from any varbind values you do not want NNMi to decode, such as binary data. Use the examples shown in the `nnmvbnosrcenc.conf` file as templates to configure your combinations. This tells NNMi to display the Custom Incident Attribute values in the Incident form using a hexadecimal value.
- c Save your changes.
- d From a command prompt, run `ovstop ovjboss`.
- e From a command prompt, run `ovstart ovjboss`.
- f Test your changes to make sure these changes result in a hexadecimal display of the formerly garbled strings.

Glossary

A

account

See [user account](#).

active cluster node

See [active server](#).

active server

The server currently running the NNMi processes in an application failover or high availability configuration.

address hint

See [discovery hint](#).

application failover

In NNMi, the optional capability (configured by the user and utilizing jboss clustering support) that transfers control of NNMi processes to a standby server if the currently active server fails.

ARP cache

The ARP (Address Resolution Protocol) cache is an operating system table that maps Data Link Layer (OSI Layer 2) addresses to Network Layer (OSI Layer 3) addresses. Data Link Layer addresses are typically MAC addresses, while Network Layer addresses are typically IP addresses. In [rule-based discovery](#), NNMi uses ARP cache entries on discovered nodes (as well as other techniques) to find additional nodes that can be checked against the current discovery rules.

auto-discovery

See [rule-based discovery](#).

C

Causal Engine

NNMi technology that applies [root cause analysis](#) (RCA) to network symptoms, using a [causality](#)-based approach. Causal Engine RCA is triggered by certain occurrences, including changes detected as a

result of [state polling](#), [SNMP traps](#), and specific [incidents](#). The Causal Engine uses RCA to determine the [status](#) of managed objects, to formulate [conclusions](#) about them, and to generate [root cause incidents](#).

causality

Denotes the relationship between one event (the cause) and another event (the effect) which is the direct consequence (result) of the first. NNMi uses causality analysis algorithms to analyze event cycles and identify solutions for resolving network issues.

cluster

In an NNMi context, a grouping of hardware and software, linked by high availability technology or by using jboss clustering capabilities, that works together to ensure functional and data continuity in the event of component overload or failure. The computers in a cluster are commonly connected to each other through high speed LANs. Clusters are usually deployed to improve availability and/or performance.

cluster member or node

In an NNMi context, a system within a high availability or jboss cluster that has been or will be configured to support NNMi high availability or application failover.

community string

A password-like mechanism used in [SNMPv1](#) and [SNMPv2c](#) implementations to authenticate SNMP queries to SNMP agents. The community string is passed in cleartext in SNMP packets, making it vulnerable to packet sniffing. [SNMPv3](#) provides stronger security mechanisms for authentication.

conclusion

In NNMi, supporting detail generated and used by the [Causal Engine](#) that sheds further light on how the Causal Engine determined [status](#) and [root cause incidents](#) for a managed object.

console

See [NNMi console](#).

controller

In NNMi application failover, a JGroups term for the cluster member that has the master cluster state. The controller is always the oldest member of the cluster.

D

discovery hint

An IP address found by NNMi using an SNMP ARP cache query; a CDP, EDP, or other discovery protocol query; or a ping sweep. NNMi further queries IP addresses found as discovery hints, then checks the results against the current discovery rules in [rule-based discovery](#).

discovery process

The process by which NNMi gathers information about network [nodes](#) so that they can be placed under management. Initial discovery runs as a two-phase process, returning device inventory information and then network connectivity information.

After initial discovery, the discovery process is ongoing. In [list-based discovery](#), this means devices in the list of seeds will be updated if their configuration changes. In [rule-based discovery](#), new devices will also be added if they match current [discovery rules](#). Discovery can also be initiated on demand for a device or set of devices from the [NNMi console](#) or from the command line.

See also [spiral discovery](#), [rule-based discovery](#), and [list-based discovery](#).

discovery rule

A range of user-defined IP addresses and/or system object IDs (OIDs) used to limit the [rule-based discovery](#) process. Configure discovery rules in the **Discovery Configuration** portion of the NNMi console under **Auto-Discovery Rules**. See also [rule-based discovery](#).

discovery seed

See [seed](#).

E

embedded database

The database included with NNMi. NNMi can also be configured to use an external Oracle database

instead of the embedded database for most of its tables. See also [PostgreSQL](#).

episode

A term used in NNMi [root cause analysis](#) to refer to a specific duration, triggered by a primary failure, during which secondary failures are suppressed or are correlated under the primary failure.

F

fault polling

A key NNMi monitoring activity, in which NNMi issues ICMP pings and/or SNMP read-only queries of status MIBs for its managed interfaces, IP addresses, and SNMP agents to determine the [state](#) of each managed object. Users can customize the types of fault polling performed for different interface groups, node groups, and nodes under **Monitoring Configuration** in the **Configuration** workspace of the [NNMi console](#). Fault polling is a subset of [state polling](#).

G

global network management

A distributed deployment of NNMi with one or more global managers consolidating data from one or more geographically distributed regional managers.

global manager

The NNMi management server in a global network management deployment that consolidates data from distributed NNMi regional manager servers. The global manager provides a unified view of topology and incidents across the whole environment. A global manager must have an NNMi Advanced license.

H

HA

See [high availability](#).

HA resource group

In modern [high availability](#) environments such as HP ServiceGuard, Veritas Cluster Server, or Microsoft Cluster Services, applications are represented as compounds of resources, such as the application itself, its shared file systems and a virtual IP address. The resources consist of an *HA resource group*, which represents an application running in a cluster environment.

high availability

Used in this guide to refer to a hardware and software configuration that provides for uninterrupted service if part of the configuration fails. High availability (HA) means that the configuration has redundant components to keep applications running at all times even if a component fails. NNMi can be configured to support one of several commercially available HA solutions. Contrast with [application failover](#).

HP Network Node Manager i Software

An HP software product (abbreviated NNMi) designed to aid network administration and to consolidate network management activities, including the ongoing discovery of network nodes, monitoring events, and network fault management. Primarily accessed from the [NNMi console](#).

HP Network Node Manager i-suite Software

Also called HP NNM i-suite Software, a family of HP software products built around the [HP Network Node Manager i Software](#) product (NNMi). HP NNM i-suite Software includes NNMi, NNMi Advanced, and associated iSPIs such as the NNM iSPI for Performance and the NNM iSPI for MPLS.

I

ICMP

See [Internet Control Message Protocol](#).

incident

In NNMi, a notification of an occurrence related to your network, displayed in [NNMi console](#) incident views and forms. NNMi includes a number of **Incident Management** and **Incident Browsing** views that enable users to filter incidents based on incident attributes. Most incident views display incidents generated directly by NNMi (sometimes called *management events*). NNMi also includes views for browsing incidents generated from [SNMP traps](#) and from [NNM 6.x/7.x events](#).

interface

A physical port used to connect a node to the network.

interface group

One of NNMi's primary filtering techniques, where interfaces are grouped together to apply settings to a group or filter visualizations by group. Interface groups can be used for any or all of the following:

configuring monitoring, filtering table views, and customizing map views. See also [node group](#).

Internet Control Message Protocol

One of the core protocols of the Internet protocol suite (TCP/IP). ICMP ping is used by NNMi along with SNMP queries for [state polling](#).

iSPI

See [NNM iSPI](#).

L

L2

See [Layer 2](#).

L3

See [Layer 3](#).

Layer 2

Refers to the Data Link Layer of the multi-layered communication model, Open Systems Interconnection (OSI). The data link layer moves data across the physical links in the network. NNMi Layer 2 views provide information about the physical connectivity of devices.

Layer 3

Refers to the Network Layer of the multi-layered communication model, Open Systems Interconnection (OSI). The network layer is concerned with knowing the address of the neighboring nodes in the network, selecting routes, and quality of service. NNMi Layer 3 views provide information about connectivity from a routing perspective.

list-based discovery

A process, based on a list of seeds, that discovers and returns detailed network information *only about the nodes that you specify as seeds*. List-based discovery maintains a limited network inventory for specific queries and tasks. Contrast with [rule-based discovery](#). See also [discovery process](#) and [spiral discovery](#).

logical volume

A computer storage virtualization term referring to an arbitrarily sized space in a [volume group](#) that can be used as a separate file system or as a device swap space. Several of the [high availability](#) products supported by NNMi use logical volumes in their shared file systems.

M

management server

The NNMi management server is the computer system on which the NNMi software is installed. The NNMi processes and services run on the NNMi management server. (Prior NNM revisions used the term “NNM management station” for this system.)

MIB

See [Management Information Base](#).

Management Information Base

In SNMP, the collection of data about the managed network, organized hierarchically. The data objects within the management information base refer to characteristics of managed devices. NNMi collects network management information by making SNMP queries to and receiving SNMP traps from managed [nodes](#) using MIB data objects (sometimes called “MIB objects,” “objects,” or “MIBs”).

N

NNM 6.x/7.x events

An NNMi term for events forwarded from older NNM management stations to NNMi. NNMi provides incident views for browsing the incidents that NNMi generates from these forwarded events.

NNM iSPI

A Smart Plug-in within the [HP Network Node Manager i-suite Software](#) family. An NNM iSPI adds functionality to NNMi for a specific technology such as MPLS or for a specific domain such as network engineering.

NNMi

See [HP Network Node Manager i Software](#).

NNMi console

The NNMi user interface. Operators and administrators use the NNMi console for network management tasks in NNMi.

node

In the network context, a computer system or device (for example, printer, router, or bridge) in a network. While nodes that are able to respond to SNMP queries provide NNMi with the most comprehensive management information, NNMi can also perform restricted management of non-SNMP nodes.

node group

One of NNMi’s primary filtering techniques, where nodes are grouped together to apply settings to a group or filter visualizations by group. Node groups can be used for any or all of the following: configuring monitoring, filtering table views, and customizing map views. See also [interface group](#).

O

OID

See [Object Identifier](#).

Object Identifier

In SNMP, a numerical sequence that identifies a [MIB](#) data object. An OID consists of numbers separated by dots in which each number represents a particular data object at that level of the MIB hierarchy. The OID is the numerical equivalent of the MIB object name, for example, the MIB object name `iso.org.dod.internet.mgmt.mib-2.bgp.bgpTraps.bgpEstablished` is equivalent to its OID `1.3.6.1.2.1.15.0.1`.

ovstatus command

A command that reports the current status of the NNMi managed processes. Can be invoked from the NNMi console (**Tools > NNMi Status**) or at a command prompt. See the *ovstatus* reference page, or the UNIX manpage.

ovstart command

A command that starts the NNMi managed processes. Invoked at a command prompt. See the *ovstart* reference page, or the UNIX manpage.

ovstop command

A command that stops the NNMi managed processes. Invoked at a command prompt. See the *ovstop* reference page, or the UNIX manpage.

P

ping sweep

A network probe technique that sends ICMP ECHO requests to multiple IP addresses to determine which addresses are assigned to responsive nodes. When enabled in [rule-based discovery](#), NNMi can use ping sweep on configured IP address ranges to find additional nodes. Some network administrators block ICMP ECHO requests because ping sweeps can be used in denial-of-service attacks.

port

In a network hardware context, a connector for passing information into and out of a network device.

PostgreSQL

An open source relational database which NNMi uses by default to store information such as topology, incidents, and configuration information. NNMi can also be configured to use Oracle instead of PostgreSQL for most of its tables.

public key certificate

Used in network security and encryption, a file that incorporates a digital signature to bind together a public key with identity information. A certificate is used to verify that a public key belongs to an individual or organization. NNMi uses SSL certificates, which contain a public key and a private key, for authentication and encryption of client/server communication.

R

RCA

See [root cause analysis](#).

region

In NNMi, a grouping of devices for the purpose of configuring communication settings such as timeout values and access credentials.

regional manager

The NNMi management server in a global network management deployment that provides discovery, polling and trap reception for devices and forwards information to the global manager.

role

See [user role](#).

rule

See [discovery rule](#).

rule-based discovery

Often called *auto-discovery*, NNMi can use rule-based discovery to seek out [nodes](#) that NNMi should add to its database, following user-specified [discovery rules](#). NNMi looks for [discovery hints](#) in data from discovered nodes, then checks these candidates against the specified discovery rules. Configure discovery rules in the **Discovery Configuration** portion of the NNMi console under

Auto-Discovery Rules. Contrast with [list-based discovery](#).

root cause analysis

In NNMi, root cause analysis (RCA) refers to a class of problem solving methods used by NNMi to determine root causes for network issues. In NNMi, the root cause is the actionable issue that will resolve associated problem symptoms if it is addressed. NNMi uses the identification of the root cause in two key ways: to notify the user of the actionable problem and to suppress reporting of secondary problem symptoms until the root cause issue has been resolved. Determination of root cause may result in status changes for managed objects, generation of [root cause incidents](#), or both.

An example of how NNMi uses RCA is the scenario in which a managed router fails, and managed nodes on the other side of the router from the NNMi [management server](#) can no longer respond to [state polling](#) queries. NNMi uses RCA to determine that the state polling failures are secondary problem symptoms. It reports the router failure as the root cause incident and refrains from reporting the problem symptoms for the downstream nodes until the root cause router failure is resolved.

root cause incident

An NNMi [incident](#) in which the *Correlation Nature* attribute is set to *Root Cause*. NNMi uses [root cause analysis](#) (RCA) to establish the root cause incident as the actionable issue that will resolve associated problem symptoms if it is addressed. See [root cause analysis](#).

S

seed

A network node that helps NNMi discover your network by acting as a starting point for the network discovery process. For example, a seed might be a core router in your management environment. Each seed is identified by an IP address or host name. Unless [rule-based discovery](#) has been configured, NNMi's discovery process is limited to [list-based discovery](#) of specified seeds.

seeded discovery

See [list-based discovery](#).

Simple Network Management Protocol

A simple protocol operating at the application layer (Layer 7) of the OSI model, by which management information for a network element can be inspected

or altered by remote users. SNMP is the predominant protocol used by NNMi to exchange network management information with agent processes on managed nodes. NNMi supports the three most common versions of SNMP: SNMPv1, SNMPv2c, and SNMPv3.

SNMP

See [Simple Network Management Protocol](#).

SNMP trap

Network management using polling (solicited responses from SNMP agents) is an SNMP design principle that promotes simplicity. However, the protocol does provide for communication of unsolicited messages from SNMP agents to the SNMP manager process (in this case, NNMi). Unsolicited agent messages are known as “traps” and are generated by SNMP agents in response to internal state changes or fault conditions. NNMi generates [incidents](#) from received SNMP traps, displayed in the **SNMP Traps** incident browsing view.

SNMP trap storm

A high number of unsolicited SNMP agent messages that can overwhelm an SNMP manager process (in this case, NNMi). You can configure SNMP trap storm thresholds in NNMi, using the `nmtrapconfig.ovpl` command. NNMi blocks traps when incoming trap rates exceed the specified threshold rate, until the trap rates fall below the re-arm rate.

spiral discovery

NNMi’s ongoing refinement of network topology information, which includes information about inventory, containment, relationships, and connectivity in networks managed by NNMi. See also [discovery process](#), [rule-based discovery](#), and [list-based discovery](#).

state

NNMi generally uses the term **state** for self-reported managed object responses related to MIB II `ifAdminStatus`, MIB II `ifOperStatus`, performance, or availability. Contrast with [status](#).

state polling

The directed monitoring performed by NNMi’s State Poller, which uses ICMP ping and SNMP queries to retrieve fault, performance, component health, and availability data from managed objects. See also [fault polling](#).

status

In NNMi, an attribute of a managed object that indicates its overall health. The status is calculated by the [Causal Engine](#) from the managed object’s outstanding [conclusions](#). Contrast with [state](#).

sysObjectID

See [system object ID](#).

system object ID

In NNMi, a specialized term for an SNMP [Object Identifier](#) that identifies a model or type of network element. The system object ID is part of a network element’s [MIB](#) object, which is queried by NNMi from individual nodes during discovery. Examples of network element types that can be classified by their system object IDs include any member of the HP ProCurve switch family, an HP J8715A ProCurve Switch, and an HP SNMP agent for HP IPF systems. Other vendors’ network elements can be likewise classified according to their system object IDs. A key use for the system object ID is in defining NNMi Device Profiles, which specify characteristics of network elements that can be deduced once a network element’s type is known.

system account

In NNMi, a special account provided for use during NNMi installation. After installation, the NNMi system account should only be used for command-line security and for recovery purposes. Contrast with [user account](#).

T

topology (network)

In communication networks, a schematic description of the arrangement of a network, including its nodes and connections.

trap

See [SNMP trap](#).

U

unconnected interface

From NNMi’s perspective, an unconnected interface is an interface that is not connected to another device discovered by NNMi. By default, the only unconnected interfaces that NNMi monitors are those that have IP addresses *and* are contained in nodes from the **Routers** node group.

user account

In NNMi, a way to provide access to NNMi for users or groups of users. NNMi user accounts are set up in the NNMi console and implement predetermined user roles. See [system account](#) and [user role](#).

user role

As part of setting up user access, the NNMi administrator assigns a pre-configured user role to each NNMi user account. User roles determine which user accounts can access the NNMi console, as well as which workspaces and actions are available to each user account. NNMi provides the following hierarchical user roles, which are predefined by the program and cannot be modified: *Administrator*, *Web Service Client*, *Operator Level 2*, *Operator Level 1*, *Guest*. See also [user account](#).

V**virtual host name**

The host name associated with a [virtual IP address](#).

virtual IP address

An IP address that is not tied to any particular network hardware, used in high availability configurations to send uninterrupted network traffic to the most appropriate server based on current failover or load-balancing needs.

volume group

A computer storage virtualization term referring to one or more disk drives that are configured to form a single large storage area. Several of the [high availability](#) products supported by NNMi use volume groups in their shared file systems.

Index

A

- access credentials
 - community strings, 47
 - configuring, 46
- accessing NNMi, 114
- actions, NA integration
 - added, 429
 - incidents, 430
 - viewing incidents, 431
- Actions menu, 376
- active
 - protocols, 52
 - server, 180
- adding NNM 6.x/7.x
 - management stations, 368
 - views, 372
- address, management server
 - changing, 288
 - preferred, 49
 - verifying, 54
- advantages
 - discovery
 - list-based, 59
 - rule-based, 59
 - integration
 - BSM Operations Management, 472
 - BSM topology, 416
 - CiscoWorks LMS, 389
 - HPOM, agent implementation, 488
 - HPOM, web services implementation, 507
 - NA, 426
 - NNMi Integration Module for Netcool Software, 526
 - NNMi northbound interface, 456
 - SIM, 444
 - UCMDB, 424
- AlarmPoint
 - agent, 384
 - client, 384
 - Java Client, 383
 - servers, 383
- AlarmPoint integration
 - AlarmPoint Mobile Gateway
 - disabling, 387
 - documentation, 386
 - enabling, 386
 - overview, 385 to 386
 - supported versions, 385
 - troubleshooting, 387
 - usage, 386
 - AlarmPoint platform
 - disabling, 385
 - documentation, 383
 - enabling, 383 to 384
 - overview, 382 to 383
 - supported versions, 383
 - troubleshooting, 385
 - usage, 384
- alarms, NNM 6.x/7.x
 - categories, 369
 - launching dynamic views, 376
 - mapping, 369 to 370
- alerting application, 381
- APAgent (AlarmPoint), 384
- APClient (AlarmPoint), 384
- application
 - alerting, 381
 - failover
 - configuring NNMi, 181 to 184
 - configuring on multi-subnets, 196
 - feature, 184 to 187
 - incidents, 187
 - integrated products, 189
 - NNM iSPIs, 187
 - scenarios, 186
 - setup, 180
 - server, AlarmPoint, 383
- Application_A.log file, 242
- Application Alert Alarms category, NNM 6.x/7.x, 370
- Application Status incident category, 370
- architecture, HA cluster, 203
- ARP cache, 63

- Asset Manager integration
 - disabling, 402
 - documentation, 402
 - enabling, 402
 - supported versions, 402
 - troubleshooting, 402
 - usage, 402
- attribute
 - Author, 38
 - Ordering, 38
- authentication failures, reducing, 55
- authentication profiles
 - multiple, 52
- Author attribute, 38
- B**
- backbone, network, 57
- backing up data
 - embedded database only, 262
 - script, 256 to 257
 - strategy, 260 to 261
- BAC My BSM initialization string, 115
- BAC My BSM integration
 - configuring demonstration portlets, 406
 - configuring single sign-on, 410
 - creating custom portlets, 407 to 410
 - creating SSO, 115 to 116
 - default modules, 404 to 405
 - documentation, 404
 - overview, 403 to 404
 - parameters, 412 to 413
 - portal description, 403
 - troubleshooting, 410
 - troubleshooting reports, 411
- Basic shared disk format
 - HA, 211
- batching monitoring configuration, 78
- behavior
 - polling defaults, 73
- behavior, configuring
 - HPOM integration, web services implementation, 521
 - NA integration, 438
- benefits
 - discovery
 - list-based, 59
 - rule-based, 59
- integration
 - BSM Operations Management, 472
 - BSM topology, 416
 - CiscoWorks LMS, 389
 - HPOM, agent implementation, 488
 - HPOM, web services implementation, 507
 - NA, 426
 - NNMi Integration Module for Netcool Software, 526
 - NNMi northbound interface, 456
 - SIM, 444
 - UCMDB, 424
- best practices
 - Author attribute, 38
 - batching monitoring configuration, 78
 - creating
 - object group definitions, 76
 - reusable node groups, 77
 - double-checking order numbers, 53
 - No Device Profile, 333
 - Ordering attribute, 38
 - overview, 23
 - preferred management address, 49
 - preparing NNMi configuration move, 283
 - saving existing configuration, 38
 - short polling intervals, 77
 - verifying order numbers, 79
- BIND, 222
- browsers, incident, 365
- BSM. *See* BSM integration
- BSM integration
 - Operations Management
 - benefits, 472
 - changing, 479
 - disabling, 480
 - documentation, 473
 - enabling, 473
 - overview, 472
 - parameters, 483 to 486
 - supported versions, 473
 - troubleshooting, 480 to 482
 - usage, 476
 - topology
 - benefits, 416
 - changing, 418
 - disabling, 418
 - documentation, 416
 - enabling, 416
 - overview, 415
 - parameters, 419 to 422
 - supported versions, 416
 - topology filter, 421
 - troubleshooting, 418
 - usage, 417

Business Availability Center. *See* BAC My BSM integration
Business Service Management. *See* BSM integration

C

cache, ARP, 63
categories, mapping, 369
certificate, public key, 101
changing
 BSM integration
 Operations Management, 479
 topology, 418
 CiscoWorks LMS integration, 392
 HPOM integration
 agent implementation, 497
 web services implementation, 513
 NA integration, 434
 NNMi Integration Module for Netcool Software, 531
 NNMi management server
 IP address, 288
 overview, 283
 NNMi northbound interface, 463
 SIM integration, 446
 UCMDB integration, 424
checklist, polling, 72
CIAs, 371
Cisco
 routers
 defining node groups, 40
 hierarchies, 41
 switches
 defining node groups, 40
 hierarchies, 41
Cisco. *See* CiscoWorks LMS integration
CiscoWorks LAN Management Solution. *See* CiscoWorks LMS integration
CiscoWorks LMS integration
 benefits, 389
 changing, 392
 description, 389
 disabling, 392
 documentation, 390
 enabling, 390 to 391
 parameters, 393 to 394
 supported versions, 390
 troubleshooting, 392
 usage, 391

ClarusIPC Plus⁺ integration
 with NNMi
 disabling, 396
 documentation, 396
 enabling, 396
 overview, 395 to 396
 supported versions, 396
 troubleshooting, 397
 usage, 396
 with NNM iSPI for IP Telephony
 disabling, 399
 documentation, 398
 enabling, 398
 overview, 397 to 398
 supported versions, 398
 troubleshooting, 399
 usage, 399
Clarus Systems ClarusIPC Plus⁺. *See* ClarusIPC Plus⁺ integration
Client, AlarmPoint Java, 383
cluster.exe command, 219
cluster.log file, 241
cluster architecture for HA, 203
cluster manager, modes for application failover, 184
Cluster Manager process, 180
Cluster Member process, 180
cluster nodes, configuring HA
 NNMi
 primary, 211 to 214
 secondary, 214
cmcluster file, 241
coexistence
 HP Operations agent, 33
 HP Performance Insight, 33
collecting data
 status polling
 choosing data, 78
 verifying, 81
command line mode, application failover, 184
command-line security, 114
commands
 cluster.exe, 219
 hostname, 222
 nnm.envvars.bat, 542 to 545
 nnm.envvars.sh, 542
 nnmcluster, 183
 nnmcommconf.ovpl, 54

- nnmconfigexport.ovpl
 - outputting configuration to XML, 284
 - saving NNM 6.x/7.x management station configuration, 369
 - SNMPv3 credentials, 38
- nnmconfigimport.ovpl, 284
- nnmdatareplicator.ovpl, 219
- nnmdumpevents, 376
- nnmhaconfigure.ovpl, 237
- nnmhadisk.ovpl
 - replicating configuration file, 218
 - troubleshooting nmsdbmgr, 237
- nnmhargconfigure.ovpl, 235
- nnmimport.bat, 434
- nnmimport.sh, 434
- nnmlicense.ovpl
 - changing management server, 288
 - licensing NNMi in HA cluster, 220 to 222
- nnmloadseeds.ovpl, 65
- nnmofficialfqdn.ovpl, 114
- nnmsetofficialfqdn.ovpl, 114
- nnmtopodump.ovpl, 359
- nslookup, 210
- ovaddr, 371
- ovspmd, 236
- ovstart
 - application failover, 184 to 187
 - troubleshooting, 236
- ovstop
 - application failover, 184 to 187
 - troubleshooting, 236
 - without causing failover, 224, 229, 231
- ovtopodump, 374
- ovtopofix, 374
- remsh, 202
- sendMsg.ovpl, 373 to 375
- snmptrap, 375
- ssh, 202
- traceroute, 52
- xnmevents, 367

command scripts, configuring NA, 430

communication

- concepts, 46
- configuration regions, 50
- configuring, 53
- evaluating configuration, 53
- planning, 50
- settings, 54
- tuning, 55
- verifying settings, 54

community strings

- access credentials, 47
- multiple, 52

Component Health monitoring, 78 to 83

component registration files, 258

concepts

- communication, 46
- configuration, 37
- discovery, 315
- event monitoring, 320
- HA, 203
- incidents, 85 to 93
- state polling, 71 to 72
- status monitoring, 318

Configuration

- incident category, 370
- workspace, 371

configuration

- batching monitoring, 78
- comparisons
 - customizing event monitoring, 318
 - network discovery, 313
 - status monitoring, 316
- concepts, 37
- files
 - HA clusters, 239
 - replication for HA, 219
- HA information
 - NNMi, 210
- incident category, 370
- levels, 46
- moving NNMi, 284
- nodes, 51
- polling example, 73
- preparing NNMi move, 283
- redoing, 44
- regions, 50
- saving
 - existing, 38
 - NNM 6.x/7.x management station, 369
- scripts, HA clusters, 239
- settings, 43
- shared data, 258
- transaction-based updates, 38
- troubleshooting HA, 234
- verifying network monitoring, 80 to 81
- workspace, 371

Configuration Alarms category, NNM 6.x/7.x, 370

Configuration workspace

- configuring state polling, 78
- evaluating state polling, 80

- configuring
 - access credentials, 46
 - application failover, 179
 - BAC My BSM
 - portlets, 406
 - communication, 53
 - HA cluster nodes, primary
 - NNMi, 211 to 214
 - HA cluster nodes, secondary
 - NNMi, 214
 - HA clusters
 - NNMi, 209 to 215
 - incidents, 94 to 95
 - interface
 - groups, 78
 - monitoring, 79
 - list-based discovery, 65
 - NA diagnostics, 430
 - NNM 6.x/7.x
 - adding views, 372
 - forwarding events, 366
 - node
 - groups, 78
 - monitoring, 79
 - retries, 46
 - rule-based discovery, 64
 - shared disk, 219
 - SNMP
 - access, 327 to 333
 - state polling, 78 to 80
 - timeouts, 46
 - configuring behavior
 - HPOM integration, web services
 - implementation, 521
 - NA integration, 438
 - connectivity, verifying network, 66
 - console. *See* NNMi console
 - container views, upgrading from NNM 6.x/7.x, 358 to 359
 - containment, node group, 41
 - controlling traffic, 52
 - CPU resources, 83
 - creating
 - NNM 6.x/7.x management station entity, 371
 - NNMi portlets, 407 to 410
 - object group definitions, 76
 - reusable node groups, 77
 - credentials, access
 - community strings, 47
 - configuring, 46
 - custom incident attributes, 371
 - customizing event monitoring, 318
 - custom scripts, upgrading from NNM 6.x/7.x, 359
- ## D
- daemon mode, application failover, 184
 - data
 - backing up
 - embedded database only, 262
 - script, 256 to 257
 - strategy, 260 to 261
 - collection, State Poller, 78
 - restoring
 - embedded database only, 262
 - script, 259
 - shared
 - configuration, 258
 - disk, 218
 - verifying collection, 81
 - data backup
 - configuration scope, 257
 - event scope, 257
 - topology scope, 257
 - database
 - backing up embedded, 262
 - Postgres, 180
 - resetting, 44
 - restoring embedded, 262
 - topology, 71
 - database, embedded
 - moving NNMi, 284
 - defaults
 - BAC My BSM modules, 404 to 405
 - community strings, 55
 - configuration settings, 43
 - discovery, 57
 - HTTPS ports, 547
 - meaningful, communication, 50
 - NA integration modifications to incidents, 430
 - polling behavior, 73
 - routers, 68
 - rule-based discovery, 64
 - switches, 68
 - verifying settings, 80
 - defining node groups, 40
 - deleting
 - discovered nodes, 66
 - license key, 288
 - destination list file, 367
 - details, node, 373

- devices
 - filters, 41
 - importing NNMi into NA, 433
 - profiles
 - concepts, 41
 - State Polling, 77
 - SNMP access, 54
 - diagnostics, configuring NA, 430
 - directories. *See* locations
 - directory service, integration NNMi with, 119
 - directory service, NNMi user information in, 119
 - disabling
 - AlarmPoint integration, 385
 - AlarmPoint Mobile Gateway integration, 387
 - Asset Manager integration, 402
 - BSM integration
 - Operations Management, 480
 - topology, 418
 - CiscoWorks LMS integration, 392
 - ClarusIPC Plus+ integration
 - with NNMi, 396
 - with NNM iSPI for IP Telephony, 399
 - HPOM integration
 - agent implementation, 498
 - web services implementation, 514
 - NA integration, 434
 - network access, 49
 - nGenius Performance Manager integration, 454
 - NNMi Integration Module for Netcool Software, 531
 - NNMi northbound interface, 463
 - PCM Plus integration, 440
 - RAMS MPLS WAN integration, 442
 - SIM integration, 446
 - SNMP, 52
 - traffic, 49
 - UCMDB integration, 424
 - disadvantages, list-based discovery, 59 to 60
 - discovery
 - approaches
 - list-based discovery, 59
 - rule-based discovery, 59 to 60
 - deleting nodes, 66
 - evaluating
 - list-based discovery, 66
 - rule-based discovery, 67 to 68
 - key concepts, 315
 - NNM 6.x/7.x
 - comparison, 313
 - upgrading from, 333 to 343
 - performance, 55
 - restarting, 44
 - routers, 67
 - spiral, 57
 - switches, 67
 - transaction-based updates, exception, 38
 - disk
 - failover, 237
 - group, HA configuration, 211
 - shared
 - copying data files, 209
 - directories, 218
 - DiskGroup_A.log file, 242
 - DNS, 222
 - documentation
 - AlarmPoint integration, 383
 - AlarmPoint Mobile Gateway integration, 386
 - Asset Manager integration, 402
 - BAC My BSM integration, 404
 - BSM integration
 - Operations Management, 473
 - topology, 416
 - CiscoWorks LMS integration, 390
 - ClarusIPC Plus+ integration
 - with NNMi, 396
 - with NNM iSPI for IP Telephony, 398
 - hardware and software requirements, NNMi, 31
 - HPOM integration
 - agent implementation, 489
 - web services implementation, 508
 - locations, 321
 - NA integration, 427
 - nGenius Performance Manager integration, 453
 - PCM Plus integration, 440
 - RAMS MPLS WAN integration, 442
 - related, 26
 - SIM integration, 444
 - UCMDB integration, 424
 - Domain Name Service, 222
 - dynamic
 - disks, 219
 - views
 - NNM 6.x/7.x, 371
 - NNMi, 376
- ## E
- editing trapd.conf, 368
 - embedded database
 - backing up and restoring, 262
 - relational, 255

- enabling
 - AlarmPoint integration, 383 to 384
 - AlarmPoint Mobile Gateway integration, 386
 - Asset Manager integration, 402
 - BSM integration
 - Operations Management, 473
 - topology, 416
 - CiscoWorks LMS integration, 390 to 391
 - ClarusIPC Plus⁺ integration
 - with NNMi, 396
 - with NNM iSPI for IP Telephony, 398
 - HPOM integration, agent implementation, 490
 - HPOM integration, web services
 - implementation
 - Linux, 510
 - UNIX, 510
 - Windows, 508
 - NA integration, 427 to 429
 - nGenius Performance Manager integration, 453
 - NNMi northbound interface, 457
 - PCM Plus integration, 440
 - RAMS MPLS WAN integration, 442
 - rule-based discovery, 67
 - SIM integration, 444 to 445
 - end nodes, avoiding managing, 57
 - environment, test, 23
 - environment variables
 - overview, 541
 - UNIX
 - administration, 544
 - installation, 24
 - Windows
 - administration, 542 to 545
 - application failover, 180
 - installation, 24
 - Error Alarms category, NNM 6.x/7.x, 370
 - /etc/hosts, 222
 - evaluating configuration
 - communication, 53
 - discovery, 66
 - incidents, 95 to 96
 - state polling, 80 to 82
 - evaluation order, 72
 - Event Configuration window, NNM 6.x/7.x, 366
 - Event Configurator window, NNM 6.x/7.x, 367, 377
 - events
 - forwarding
 - management server, 366
 - verifying, 373
 - generating test interface down/up events, 374
 - monitoring
 - concepts, 320
 - customizing, 318
 - OV_Message, NNM 6.x/7.x, 375
 - reducing, 368
 - upgrading from NNM 6.x/7.x, 349 to 356
 - examples
 - configuring node groups, 79
 - container view configuration, 359
 - HPOM integration, web services
 - implementation
 - incident filters, 523
 - polling configuration, 73
 - SNMP information, 328
 - excluding devices from discovery, 60
 - ext2 shared disk format
 - HA, 211
 - extended monitoring, 73
 - external relational database, 255
- ## F
- failover
 - application
 - configuring NNMi, 181 to 184
 - configuring on multi-subnets, 196
 - feature, 184 to 187
 - integration products, 189
 - NNM iSPIs, 187
 - scenarios, 186
 - setup, 180
 - disk, 237
 - failure, reducing authentication, 55
 - files
 - backing up, 255
 - component registration, 258
 - destination list, 367
 - HA configuration
 - files, 239
 - scripts, 239
 - ipNoLookup.conf, 331
 - jbossServer.log, 237
 - ldap.properties, 140

- license.txt
 - active cluster node, 222
 - management server, 288
- LicFile.txt
 - backing up, 212
 - HA cluster, 220
 - location, 258
- lwssofmconf.xml, 113, 115, 153
- netmon.cmstr, 328
- netmon.noDiscover, 340
- nGeniusNNM8.zip, 453
- nmsdbmgr.log, 241
- nnmcontainerlist.csv, 359
- nnmdatatoreplicator.conf, 239
- NOC_Demo_Portal_iSPIPerf.xml, 404 to 406
- NOC_Demo_Portal.xml, 404 to 406
- not updating on cluster node, 234
- oid_to_sym, 333
- OLDsyslog.log, 241
- ov.conf
 - HA configuration, 239
 - troubleshooting nmsdbmgr, 237
- ovspmd.log, 241
- postgres.log, 241
- recovery.conf, 185
- replication, 219
- <resource_group>.cntl.log, 241
- shared disk, 237
- snmpcapture.out, 328, 359
- snmpout.txt, 328
- syslog.log, 241
- system type
 - NNMi, 210
- tar, 259
- trapd.conf, 368
- version information, 258
- XML, 284
- file system type, HA configuration, 211
- filtering
 - interface groups, 43
 - node groups, 39
- filters
 - configuring node level, 368
 - device, 41
 - incident, 522 to 524
- firewalls
 - controlling traffic, 52
 - disabling network access, 49
 - troubleshooting HPOM integration, web services implementation, 518
- flow model, task, 37
- forms
 - Incident Configuration
 - verifying NNM 6.x/7.x incident configuration, 369
 - Interface Group, 75
 - Interface Settings, 83
 - Management Stations, 371
 - Monitoring Configuration
 - refining state polling, 71
 - tuning state polling, 83
 - Node Group, 75
 - Node Settings, 83
- forwarded incidents, troubleshooting
 - BSM Operations Management, 480 to 482
 - HPOM, agent implementation, 498 to 501
 - HPOM, web services implementation, 514 to 517
 - NNMi Integration Module for Netcool Software, 532 to 533
- FORWARD field, NNM 6.x/7.x event configuration, 368
- forwarding, event
 - NNM 6.x/7.x to NNMi management server, 366
 - verifying, 373
- forwarding incidents
 - to BSM Operations Management, 472
 - to HPOM
 - agent implementation, 488
 - web services implementation, 506
 - to Netcool/OMNIbus, 526
 - to SNMP trap receiver, 456
- FQDN, 114

G

- generating test interface up/down events, 374
- getting public key certificate, 101
- group
 - disk
 - NNMi for HA, 211
 - volume
 - NNMi for HA, 211
- groups
 - interface
 - filtering, 43
 - purpose, 39
 - nodes, 39
- GUI. *See* NNMi console

H

- HA_nnmhaserver.log file, 241

- HA clusters
 - architecture, 203
 - changing hostnames
 - NNMi, 221 to 224
 - changing IP addresses
 - NNMi, 221 to 224
 - concepts, 203
 - configuration
 - files, 239
 - information, NNMi, 210
 - manpages, 208
 - reference pages, 208
 - scripts, 239
 - configuring
 - NNMi, 209 to 215
 - description, 201
 - licenses, 220
 - maintaining
 - add-on NNM iSPIs, 224
 - NNMi, 221 to 224
 - maintenance mode, 221
 - patching
 - NNMi, 229
 - prerequisites, 202
 - scenario, 205
 - shared data, 218 to 219
 - startup problems
 - nmsdbmgr, 237
 - NNMi, 236
 - pmd, 237
 - supported products, 202
 - terms, 204
 - troubleshooting configuration, 234
 - unconfiguring
 - NNMi, 225 to 228
 - upgrading NNMi, 230 to 233
- HA configuration, 239
- haconfigure.log file, 241
- hardware
 - required, 29
 - supported, 31
- HA resource group
 - cannot start, 235
 - configuration
 - NNMi, 210
 - description, 204
 - stopping
 - NNMi, 227
- hierarchies, node group, 41
- high availability clusters. *See* HA clusters
- hints
 - IP address ranges, 67
 - system ID ranges, 68
- Home Base, NNM 6.x/7.x, 376
- host, virtual for HA configuration
 - NNMi, 210
- hostname command, 222
- hostnames, changing for HA
 - NNMi, 221 to 224
- HP ServiceGuard
 - HA resource group, 204
- HP Asset Manager. *See* Asset Manager integration
- HP BSM. *See* BSM integration
- HP Business Availability Center. *See* BAC My BSM integration
- HP Business Service Management. *See* BSM integration
- HP Network Automation. *See* NA integration
- HP NNM iSPI for Metrics. *See* NNM iSPIs
- HP NNM iSPI NET. *See* NNM iSPIs
- HPOM integration
 - agent implementation
 - agent installation order, 33
 - benefits, 488
 - changing, 497
 - disabling, 498
 - documentation, 489
 - enabling, 490
 - overview, 488
 - parameters, 502 to 505
 - supported versions, 489
 - troubleshooting, 498 to 501
 - usage, 494
 - web services implementation
 - behavior, 521
 - benefits, 507
 - changing, 513
 - disabling, 514
 - documentation, 508
 - enabling, Linux, 510
 - enabling, UNIX, 510
 - enabling, Windows, 508
 - incident synchronization, 507
 - overview, 506
 - parameters, 519 to 524
 - troubleshooting, 514 to 518
 - usage, 512
- HP Operations Manager. *See* HPOM integration
- HP Operations Manager i. *See* BSM Operations Management integration
- HP Performance Insight, 33
- HP ProCurve Manager Plus. *See* PCM Plus integration

- HP Route Analytics Management System. *See* RAMS MPLS WAN integration
- HP ServiceGuard
 - HP-UX, 202
 - Linux, 202
- HP SIM. *See* SIM integration
- HP Systems Insight Manager. *See* SIM integration
- HP Universal Configuration Management Database. *See* UCMDB integration
- HP-UX
 - active server, 180
 - configuring shared disks, 219
 - HP ServiceGuard, 202
 - log files, 241
 - NIS, 222
 - nnmharg.ovpl script, 240
 - shared disk format
 - Ha, 211
 - virtual hosts, 210
- HTTPS protocol
 - default port, 547
- I**
- IBM Tivoli Netcool/OMNIbus. *See* NNMi Integration Module for Netcool Software
- ICMP
 - address monitoring, 77
 - disabling traffic, 49
 - protocol, 45
 - requests, 55
 - State Poller, 49
- identifying Layer 2 connections with mismatched states, 431
- importing NNMi devices into NA inventory, 433
- Incident Configuration form
 - verifying configurations, 369
- incidents
 - browsers, 365
 - categories, 370
 - concepts, 85 to 93
 - configuring, 94 to 95
 - evaluating configuration, 95 to 96
 - filters, 522 to 524
 - forwarding to BSM Operations Management, 472
 - forwarding to HPOM
 - agent implementation, 488
 - web services implementation, 506
 - forwarding to Netcool/OMNIbus, 526
 - forwarding to SNMP trap receiver, 456
 - planning, 93 to 94
 - synchronization of updates
 - HPOM web services implementation, 507
 - tuning, 96
 - verifying NNM 6.x/7.x in NNMi console, 369
 - viewing NA actions, 431
- Incidents workspace, 373
- Incident Web Service, HPOM, 506
- initialization string
 - BAC My BSM, 115
 - NNMi, 115
- initiating status poll, 81
- installing
 - AlarmPoint Java Client, 383
 - HP Performance Insight, 33
 - Java Runtime Environment, 370
 - license for HA cluster, 220
 - public key certificates, 101
- integrating AlarmPoint Mobile Gateway with NNMi
 - disabling, 387
 - enabling, 386
 - overview, 385 to 386
 - troubleshooting, 387
 - usage, 386
- integrating AlarmPoint with NNMi
 - disabling, 385
 - enabling, 383 to 384
 - overview, 382 to 383
 - troubleshooting, 385
 - usage, 384
- integrating Asset Manager with NNMi
 - disabling, 402
 - enabling, 402
 - troubleshooting, 402
 - usage, 402
- integrating BAC with NNMi, My BSM
 - configuring demonstration portlets, 406
 - configuring single sign-on, 410
 - configuring SSO, 115 to 116
 - creating new portlets, 407 to 410
 - default modules, 404 to 405
 - documentation, 404
 - overview, 403 to 404
 - parameters, 412 to 413
 - troubleshooting, 410

- integrating BSM with NNMi
 - Operations Management
 - benefits, 472
 - changing, 479
 - disabling, 480
 - documentation, 473
 - overview, 472
 - parameters, 483 to 486
 - troubleshooting, 480 to 482
 - usage, 476
 - topology
 - benefits, 416
 - changing, 418
 - disabling, 418
 - documentation, 416
 - enabling, 416
 - overview, 415
 - parameters, 419 to 422
 - supported versions, 416
 - topology filter, 421
 - troubleshooting, 418
 - usage, 417
- integrating CiscoWorks LMS with NNMi
 - benefits, 389
 - changing, 392
 - description, 389
 - disabling, 392
 - documentation, 390
 - enabling, 390 to 391
 - parameters, 393 to 394
 - supported versions, 390
 - troubleshooting, 392
 - usage, 391
- integrating ClarusIPC Plus⁺
 - with NNM iSPI for IP Telephony
 - disabling, 399
 - enabling, 398
 - overview, 397 to 398
 - troubleshooting, 399
 - usage, 399
 - with NNMi
 - disabling, 396
 - enabling, 396
 - overview, 395 to 396
 - troubleshooting, 397
 - usage, 396
- integrating HPOM with NNMi
 - agent implementation
 - benefits, 488
 - changing, 497
 - disabling, 498
 - documentation, 489
 - overview, 488
 - parameters, 502 to 505
 - troubleshooting, 498 to 501
 - usage, 494
 - web services implementation
 - benefits, 507
 - changing, 513
 - disabling, 514
 - documentation, 508
 - enabling, 510
 - incident synchronization, 507
 - integration behavior, 521
 - overview, 506
 - parameters, 519 to 524
 - troubleshooting, 514 to 518
 - usage, 512
- integrating NA with NNMi
 - benefits, 426
 - changing, 434
 - disabling, 434
 - documentation, 427
 - enabling, 427 to 429
 - overview, 425 to 427
 - parameters, 437 to 438
 - troubleshooting, 434
 - usage, 429 to 434
- integrating Netcool/OMNIbus with NNMi
 - benefits, 526
 - changing, 531
 - disabling, 531
 - documentation, 527
 - overview, 525
 - parameters, 534 to 538
 - supported versions, 526
 - troubleshooting, 532 to 533
 - usage, 530
- integrating nGenius Performance Manager with NNMi
 - disabling, 454
 - enabling, 453
 - troubleshooting, 452 to 454
 - usage, 454
- integrating NNM 6.x/7.x with NNMi
 - event forwarding, 366 to 370
 - remote view launching, 370 to 373
 - testing, 373 to 376
 - troubleshooting, 376

- integrating PCM Plus with NNMi
 - disabling, 440
 - enabling, 440
 - troubleshooting, 440
 - usage, 440
- integrating RAMS MPLS WAN with NNMi
 - disabling, 442
 - enabling, 442
 - parameters, 442
 - troubleshooting, 442
 - usage, 442
- integrating SIM with NNMi
 - benefits, 444
 - changing, 446
 - description, 444
 - disabling, 446
 - documentation, 444
 - enabling, 444 to 445
 - parameters, 447 to 449
 - supported versions, 444
 - troubleshooting, 447
 - usage, 446
- integrating SNMP trap receiver with NNMi
 - benefits, 456
 - changing, 463
 - disabling, 463
 - documentation, 457
 - enabling, 457
 - overview, 456
 - parameters, 466 to 470
 - troubleshooting, 463
 - usage, 459
- integrating UCMDB with NNMi
 - benefits, 424
 - changing, 424
 - description, 423
 - disabling, 424
 - documentation, 424
 - parameters, 424
 - supported versions, 424
 - troubleshooting, 424
 - usage, 424
- integration module, HPOM, 506
- integrations
 - application failover, 189
- interactive mode, application failover, 184
- interface
 - configuring monitoring, 79
 - generating test down/up events, 374
 - groups
 - configuring, 78
 - filtering, 43
 - preconfigured, 76
 - purpose, 39
 - verifying, 80
 - model, 38
 - monitoring, 73
 - settings, 43
 - virtual host network for HA configuration NNMi, 210
- Interface Group form, 75
- Interface Groups workspace, 75
- Interface Settings form, 83
- inventory, importing NNMi devices into NA, 433
- IP address
 - changing for HA NNMi, 221 to 224
 - changing NNMi management server, 288
 - ranges, 64 to 67
- ipNoLookup.conf file, 331
- IWS, HPOM, 506
- J**
- Java Client, AlarmPoint, 383
- Java Plug-in for NNM 6.x/7.x Dynamic Views, 370
- Java Runtime Environment, installing, 370
- jboss
 - application server, 180
 - directories, 258
- jbossServer.log file
 - HP-UX, 241
 - Linux, 241
 - Solaris, 242
 - UNIX, 237
 - Windows
 - configuring HA, 241
 - troubleshooting pmd, 237
- K**
- key, public, 101
- Key Operations Map portlet, 405
- L**
- lag time, testing, 52
- latency, network, 46

- Launcher, NNM 6.x/7.x, 376
- launching NNM 6.x/7.x dynamic views, 376
- Layer 2 connections, identifying mismatched states, 431
- LDAP, integrating NNMi with, 119
- ldap.properties file, 140
- levels, configuration, 46
- license.txt file
 - active cluster node, 222
 - management server, 288
- license key, NNM iSPI NET, 431
- licenses
 - capacity, 58
 - directory locations, 258
 - HA clusters, 220
 - limitations, 60
 - purchasing additional node pack, 66
- LicFile.txt file
 - backing up
 - NNMi data, 258
 - primary cluster node, 212
 - updating licenses.txt, 220
- limited environment, 23
- limiting name resolution, 331
- Linux
 - configuring shared disks, 219
 - HP ServiceGuard, 202
 - NIS, 222
 - nnmharg.ovpl, 240
 - shared disk format
 - HA, 211
 - virtual hosts, 210
- list-based discovery, 66
 - evaluating, 66
 - overview, 59
- locations
 - documentation, 321
 - AlarmPoint integration, 383
 - AlarmPoint Mobile Gateway integration, 386
 - Asset Manager integration, 402
 - BAC My BSM integration, 404
 - BSM integration, Operations Management, 473
 - BSM integration, topology, 416
 - CiscoWorks LMS integration, 390
 - ClarusIPC Plus+ integration with NNMi, 396
 - ClarusIPC Plus+ integration with NNM iSPI for IP Telephony, 398
 - HPOM integration, agent implementation, 489
 - HPOM integration, web services implementation, 508
 - NA integration, 427
 - nGenius Performance Manager integration, 453
 - PCM Plus integration, 440
 - RAMS MPLS WAN integration, 442
 - SIM integration, 444
 - UCMDB integration, 424
- environment variables
 - administration, 542 to 545
 - installation, 24
- ipNoLookup.conf, 331
- jboss deployment directory, 258
- LicFile.txt file, 258
- netmon.cmstr, 328
- netmon.noDiscover, 340
- nGeniusNNM8.zip, 453
- SNMP MIB, 258
- trapd.conf, 368

- log files
 - HA cluster
 - not updating, 234
- lvm2 shared disk format
 - HA, 211
- lwssofmconf.xml file, 113, 115, 153

M

- maintaining HA clusters, 221
- maintenance mode for HA, 221
- management
 - address
 - preferred, 49
 - verifying, 54
 - server
 - forwarding NNM 6.x/7.x events, 366
 - installing AlarmPoint Java Client, 383
 - moving NNMi, 283
 - station
 - adding NNM 6.x/7.x to topology, 368
 - creating NNM 6.x/7.x entities, 371
 - saving configuration, 369
- Management Stations form, 371
- manpages, HA, 208
- manually editing trapd.conf, 368
- mapping categories, 369
- meaningful defaults for communication configuration, 50
- membership, node group, 41

- memory resources, 83
- message browser, troubleshooting HPOM, web services implementation, 517
- messages* log files
 - Linux, 241
 - Solaris, 242
- MIB browser
 - example URL, 372
 - NNM 6.x/7.x, 376
- MIB II variables, 78
- Microsoft Cluster Services
 - dynamic disks, 219
 - HA cluster, 202
 - HA resource group, 204
 - nmhamscs.vbs script, 240
- Microsoft Failover Clustering
 - dynamic disks, 219
 - HA cluster, 202
 - HA resource group, 204
- migration. *See* upgrading
- mismatched states, Layer 2 connections, 431
- model
 - task flow, 37
 - user interface, 38
- modes, cluster manager for application failover, 184
- monitoring
 - configuration
 - batching, 78
 - settings, 43
 - verifying, 80 to 81
 - configuring node, 79
 - events
 - concepts, 320
 - customizing, 318
 - extended, 73
 - interface, 79
 - networks
 - nodes, 83
 - verifying settings, 80
 - settings, 81
 - status
 - concepts, 318
 - configuring, 316
- Monitoring Configuration form
 - description, 71
 - setting polling types and intervals, 75
 - tuning status polling, 83
- Mount_A.log file, 242
- mount point
 - NNMi, 211

- moving NNMi
 - configuration, 284
 - configuration and embedded database, 284
 - management server, 283
- MTTR, 452
- My BSM. *See* BAC My BSM integration

N

- NA, importing NNMi devices into inventory, 433
- NA integration
 - behavior, 438
 - benefits, 426
 - changing, 434
 - disabling, 434
 - documentation, 427
 - enabling, 427 to 429
 - NNMi connector, 427
 - overview, 425 to 427
 - parameters, 437 to 438
 - supported versions, 426
 - troubleshooting, 434
 - URL actions, 429
 - usage, 429 to 434
- name resolution, limiting, 331
- netmask, virtual host for HA configuration
 - NNMi, 210
- netmon.cmstr file, 328
- netmon.noDiscover file, 340
- NetScout Systems nGenius Performance Manager. *See* nGenius Performance Manager integration
- Network Automation. *See* NA integration
- Network Information Service, 222
- network interface, virtual host for HA configuration
 - NNMi, 210
- network latency, 46
- network operations center, 329
- networks
 - backbones, 57
 - disabling access, 49
 - discovering
 - evaluating, 66 to 68
 - NNM 6.x/7.x, 313
 - loads, 83
 - monitoring
 - behavior, 73
 - nodes, 83
 - settings, 80
 - verifying
 - connectivity, 66
 - settings, 80

- Network Status portlet, 405
- nGeniusNNM8.zip file, 453
- nGenius Performance Manager integration
 - disabling, 454
 - documentation, 453
 - enabling, 453
 - supported versions, 453
 - troubleshooting, 452 to 454
 - usage, 454
- NIS, 222
- nmsdbmgr.log file
 - HP-UX, 241
 - Linux, 241
 - Solaris, 242
 - Windows, 241
- nmsdbmgr service
 - disk failover, 237
 - startup problems, 237
- NNM 6.x/7.x
 - collecting data for NNMi, 325 to 327
 - configuring event node list, 368
 - customizing event monitoring, 318
 - dynamic views, 371
 - integrating with NNMi
 - event forwarding, 366 to 370
 - remote view launching, 370 to 373
 - testing, 373 to 376
 - troubleshooting, 376
 - management station, 373 to 375
 - network discovery, 313
 - status monitoring, 316
 - upgrading to NNMi
 - custom scripts, 359
 - discovery, 333 to 343
 - events, 349 to 356
 - Home Base container views, 358 to 359
 - options, 322 to 324
 - OVW maps, 356 to 358
 - phases, 322
 - SNMP, 327 to 333
 - status monitoring, 343 to 349
 - suggested path, 297, 299, 307, 321
 - views, 370
- NNM 6.x/7.x Events incident view, 376
- nnm.envvars.bat command, 542 to 545
- nnm.envvars.sh command, 542
- nnmbackup.ovpl
 - description, 255
- nnmbackupembdb.ovpl
 - description, 255
- nnmcluster command, 183
- nnmcommconf.ovpl command, 54
- nnmconfigexport.ovpl
 - outputting configuration to XML, 284
 - saving NNM 6.x/7.x management station configuration, 369
 - SNMPv3 credentials, 38
- nnmconfigimport.ovpl command, 284
- nnmcontainerlist.csv file, 359
- nnmdatareplicator.conf file, 239
- nnmdatareplicator.ovpl
 - command, 219
 - script, 240
- nnmdumpevents command, 376
- nnmhaclusterinfo.ovpl script, 239
- nnmhaconfigure.ovpl
 - command, 237
 - script, 239
- nnmhadisk.ovpl command
 - replicating configuration file, 218
 - troubleshooting, 237
 - troubleshooting nmsdbmgr, 237
- nnmhadisk.ovpl script, 240
- nnmhamonitor.ovpl script, 240
- nnmhamscs.vbs script, 240
- nnmharg.ovpl script, 240
- nnmhargconfigure.ovpl
 - command, 235
 - script, 240
- nnmhastart.ovpl script, 240
- nnmhastarttrg.ovpl script, 240
- nnmhastop.ovpl script, 240
- nnmhastoprg.ovpl script, 240
- nnmhaunconfigure.ovpl script, 239
- NNMi console
 - accessing from BSM Operations Management, 472
 - accessing from HPOM
 - agent implementation, 489
 - web services implementation, 507
 - accessing from Netcool event viewers, 526
 - signing off, 113
 - transaction-based updates, 38
 - URL, 407
 - verifying NNM 6.x/7.x incident configuration, 369
- NNMi initialization string, 115

- NNMi Integration Module for Netcool Software
 - benefits, 526
 - changing, 531
 - disabling, 531
 - documentation, 527
 - overview, 525
 - parameters, 534 to 538
 - troubleshooting, 532 to 533
 - usage, 530
- nnmimport.bat command, 434
- nnmimport.sh command, 434
- NNMi northbound interface
 - benefits, 456
 - changing, 463
 - disabling, 463
 - documentation, 457
 - enabling, 457
 - overview, 456
 - parameters, 466 to 470
 - troubleshooting, 463
 - usage, 459
- NNMi Quick Start Configuration wizard, 23
- NNMi roles in a directory service, 119
- NNM iSPI for Metrics. *See* NNM iSPIs
- NNM iSPI NET. *See* NNM iSPIs
- NNM iSPIs
 - application failover, 187
 - URL for NNM iSPI for Performance report, 407
- nnmlicense.ovpl command
 - changing management server, 288
 - licensing NNMi in HA cluster, 220 to 222
 - NNM product, 220
- nnmloadseeds.ovpl command, 65
- nnmofficialfqdn.ovpl command, 114
- nnmresetembdb.ovpl
 - description, 256
- nnmrestore.ovpl
 - description, 255
- nnmrestoreembdb.ovpl
 - description, 255
- nnmsetofficialfqdn.ovpl command, 114
- nnmtopodump.ovpl command, 359
- NOC, 329
- NOC_Demo_Portal_iSPIPerf.xml file, 404 to 406
- NOC_Demo_Portal.xml file, 404 to 406
- Node Group form, 75
- node groups
 - configuring, 78
 - defining, 40
 - device filters, 41
 - filtering, 39
 - hierarchies, 41
 - interface groups, 43
 - membership, 41
 - Non-SNMP Devices, 74
 - preconfigured, 77
 - status, 42
 - verifying, 80
- Node Group Status portlet, 405
- Node Groups workspace, 75
- nodes
 - configuration, 51
 - configuring monitoring, 79
 - configuring primary for HA clusters
 - NNMi, 211 to 214
 - configuring secondary for HA clusters
 - NNMi, 214
 - deleting discovered, 66
 - details, 373
 - filtering levels, 368
 - monitoring, 83
 - settings, 43
 - SNMP configuration, 53
- Node Settings form, 83
- Nodes inventory view, 368
- No Device Profile, 333
- Non-SNMP Devices node group, 74
- Nortel
 - routers
 - defining node groups, 40
 - hierarchies, 41
 - switches
 - defining node groups, 40
 - hierarchies, 41
- northbound interface. *See* NNMi northbound interface
- nslookup command, 210

O

- object groups, creating definitions, 76
- offline backups, 256
- oid_to_sym file, 333
- OLDsyslog.log file, 241
- om.hp.ov.nms.cluster* parameters, 185
- OM. *See* HPOM integration
- OMi. *See* BSM Operations Management integration

- online backups, 256
- OpenView Application Server, NNM 6.x/7.x, 371
- Operations Manager. *See* HPOM integration
- Operations Manager i. *See* BSM Operations Management integration
- options, upgrading from NNM 6.x/7.x, 322 to 324
- Oracle Database and HA, 216
- order, evaluation, 72
- Ordering attribute
 - auto-discovery rules, 60
 - best practices, 38
- ordering auto-discovery rules, 60
- order numbers, verifying, 79
- OV_Message event, NNM 6.x/7.x, 375
- ov.conf file
 - HA configuration, 239
 - troubleshooting nmsdbmgr, 237
- ovaddr command, NNM 6.x/7.x, 371
- ovalarm example URL, 372
- OV Application Server, NNM 6.x/7.x, 371
- ovas, NNM 6.x/7.x, 371
- ovdumpevents command, NNM 6.x/7.x, 376
- ovet*.log file
 - HP-UX, 241
- OV jovw example URL, 372
- OV Launcher example URL, 372
- ovspmd.log file
 - HP-UX, 241
 - Linux, 241
 - Solaris, 242
 - Windows, 241
- ovspmd service, troubleshooting, 236
- ovstart command
 - application server, 184 to 187
 - troubleshooting, 236
- ovstop command
 - application failover, 184 to 187
 - troubleshooting, 236
 - without causing failover
 - stopping NNMi in HA cluster, 224, 229
 - upgrading NNMi to 8.11 under HA, 231
- ovtopodump command, NNM 6.x/7.x, 374
- ovtopofix command, NNM 6.x/7.x, 374
- ovw, NNM 6.x/7.x, 376
- OVW, upgrading from NNM 6.x/7.x, 356 to 358
- ovwId, node details using, 373

P

- pages, HA reference, 208
- parameters
 - application failover, 185
 - BAC My BSM integration, 412 to 413
 - BSM integration
 - Operations Management, 483 to 486
 - topology, 419 to 422
 - CiscoWorks LMS integration, 393 to 394
 - HPOM integration
 - agent implementation, 502 to 505
 - web services implementation, 519 to 524
 - NA integration, 437 to 438
 - NNMi Integration Module for Netcool Software, 534 to 538
 - NNMi northbound interface, 466 to 470
 - RAMS MPLS WAN integration, 442
 - SIM integration, 447 to 449
 - UCMDB integration, 424
- partial backups, 257
- password, permanent license, 222
- passwords in a directory service, 119
- paths, upgrading from NNM 6.x/7.x, 297, 299, 307, 321
- PCM Plus integration
 - disabling, 440
 - documentation, 440
 - enabling, 440
 - supported versions, 440
 - troubleshooting, 440
 - usage, 440
- PCs, avoiding managing, 57
- performance, status polling, 81
- Performance incident category, 370
- Performance Insight, 33
- PERL5LIB, environment variable, 356
- permanent license password, 222
- personal computers, avoiding managing, 57
- phases, upgrading from NNM 6.x/7.x, 322
- ping protocol
 - discovering devices, 45
 - requests, 78
 - sweep, 61
- planning
 - communication, 50
 - incidents, 93 to 94
 - polling intervals, 77
 - state polling, 72 to 78
- pmd startup problems, 237

- point, mount
 - NNMi, 211
- polling
 - checklist, 72
 - configuration example, 73
 - default behavior, 73
 - planning intervals, 77
 - protocols, 49
 - status
 - evaluating performance, 81
 - initiating, 81
 - tuning, 82 to 83
- portal, BAC My BSM, description, 403
- portlets, BAC My BSM
 - configuring, 406
 - creating NNMi, 407 to 410
 - troubleshooting NNMi, 410
- port number, web server, 372
- postgres.log file
 - HP-UX, 241
 - Linux, 241
 - Solaris, 242
 - Windows, 241
- Postgres database, 180
- preconfigured
 - interface groups, 76
 - node groups, 77
- preferences
 - management address, 49
 - SNMP version, 48
- preparing NNMi configuration move, 283
- prerequisites
 - hardware, 31
 - NNMi for HA cluster, 202
 - software, 31
- preventative data backups, 261
- primary HA cluster node
 - configuration information
 - NNMi, 210
 - configuring
 - NNMi, 211 to 214
- printers, avoiding managing, 57
- problems, HA startup
 - nmsdbmgr, 237
 - NNMi, 236
 - pmd, 237
- processes
 - Cluster Manager, 180
 - Cluster Member, 180
- products, HA, 202
- profiles, authentication
 - multiple, 52
- profiles, device
 - concepts, 41
 - State Polling, 77
- protocol
 - active, 52
 - communication, 45
 - polling, 49
- public key certificate, installing, 101
- purchasing additional node pack licenses, 66

Q

- Quick Start Configuration wizard, 23

R

- RAMS MPLS WAN integration
 - disabling, 442
 - documentation, 442
 - enabling, 442
 - parameters, 442
 - supported versions, 441
 - troubleshooting, 442
 - usage, 442
- ranges, IP addresses, 67
- recovery.conf file, 185
- redoing configuration, 44
- reducing
 - authentication failures, 55
 - default community strings, 55
 - events, 368
- re-enabling NNMi for HA clusters, 236
- reference pages, HA, 208
- regions, configuration, 50
- related documentation, 26
- relational database, backing up, 255
- release notes, 31
- removing license key, 288
- remsh command, 202
- replication, configuration file, 219
- Report Presenter example URL, 372
- reports, troubleshooting BAC My BSM, 411
- requirements, hardware and software, 31
- resetting configuration, 44
- resolution, limiting name, 331

- <resource_group>.cntl.log file
 - HP-UX, 241
 - Linux, 241
 - resource group, HA
 - NNMi, 210
 - resources, system, 83
 - restarting
 - after HA maintenance
 - NNMi, 224
 - discovery, 44
 - restoring data
 - embedded database, 262
 - script, 259
 - retries
 - configuring, 46
 - tuning, 55
 - values, 52
 - routers
 - defaults, 68
 - defining node groups, 40
 - discovering, 67
 - hierarchies, 41
 - monitoring, 73
 - rule-based discovery
 - enabling, 67
 - evaluating, 67 to 68
 - overview, 59
 - rules, auto-discovery
 - evaluating, 67 to 68
 - ordering, 60
- S**
- saving NNM 6.x/7.x management station
 - configuration, 369
 - scenarios
 - application failover, 186
 - HA cluster, 205
 - scope
 - topology, 257
 - scripts
 - backup, 256 to 257
 - HA configuration, 239
 - nmhaclusterinfo.ovpl, 239
 - restoring data, 259
 - upgrading from NNM 6.x/7.x, 359
 - secondary HA cluster nodes
 - configuration information, 209
 - configuring
 - NNMi, 214
 - seeds, rule-based discovery, 59
 - sendMsg.ovpl command, NNM 6.x/7.x, 373 to 375
 - servers
 - active, 180
 - AlarmPoint, 383
 - jboss, 180
 - Service Level Agreements, 77
 - settings
 - communication, 54
 - configuration, 43
 - meaningful defaults for communication, 50
 - monitoring, 81
 - verifying default, 80
 - setting up application failover, 180
 - shared configuration data, 258
 - shared disk
 - configuring, 219
 - copying data files, 209
 - data, 218
 - files, 237
 - support, 202
 - unmounting
 - NNMi, 227
 - shared file system type, HA configuration
 - NNMi, 210
 - shared HA data, 218 to 219
 - short name, virtual host for HA configuration
 - NNMi, 210
 - SIM integration
 - benefits, 444
 - changing, 446
 - description, 444
 - disabling, 446
 - documentation, 444
 - enabling, 444 to 445
 - parameters, 447 to 449
 - supported versions, 444
 - troubleshooting, 447
 - usage, 446
 - single sign-on. *See* SSO
 - SLAs, 77
 - SNMP
 - communication problems, 66
 - community strings, 47
 - Component Health, 78
 - configuring
 - access, 327 to 333
 - Data Presenter, 372
 - device access, 54
 - disabling
 - communication, 52
 - traffic, 49

- MIB directories, 258
- monitoring, 78
- node configuration, 53
- protocol, 45
- requests, 55
- tuning settings, 55
- upgrading to NNMi, 327 to 333
- version preferences, 48
- snmpout.txt file, 328
- snmptrap command, NNM 6.x/7.x, 375
- software
 - requirements, 29
 - supported, 31
- Solaris
 - configuring shared disks, 219
 - NIS+, 222
 - nmmharg.ovpl script, 240
 - nmmhargconfigure.ovpl
 - command, 235
 - script, 240
 - shared disk format
 - Ha, 211
 - Veritas Cluster Server, 202
 - virtual hosts
 - NNMi, 210
- spiral discovery, 57
- SPIs. *See* NNM iSPIs
- ssh command, 202
- SSO
 - access, 114
 - creating for BAC My BSM integration, 115 to 116
- standby system, 180
- starting
 - after HA maintenance
 - NNMi, 224
 - HA resource group, 235
- startup problems
 - nmsdbmgr, 237
 - NNMi, 236
 - pmd, 237
- State Poller
 - concepts, 71 to 72
 - configuring, 78 to 80
 - evaluating configuration, 80 to 82
 - ICMP, 49
 - planning, 72 to 78
 - tuning, 82 to 83
 - verifying
 - communication settings, 54
 - health statistics, 82
- states, identifying Layer 2 connection mismatched, 431
- status
 - incident category, 370
 - monitoring
 - concepts, 318
 - configuring, 316
 - upgrading from NNM 6.x/7.x, 343 to 349
 - poll
 - evaluating performance, 81 to 82
 - initiating, 81
- Status Alarms category, NNM 6.x/7.x, 370
- Status incident category, 370
- stopping
 - HA resource group
 - NNMi, 227
 - without causing HA failover
 - NNMi, 224
- strategy, backup, 260 to 261
- strings, community
 - access credentials, 47
 - multiple, 52
- supported versions
 - HA products, 202
 - hardware and software, 31
 - integrated applications
 - AlarmPoint, 383
 - AlarmPoint Mobile Gateway, 385
 - Asset Manager, 402
 - BAC My BSM, 404
 - BSM Operations Management, 473
 - BSM topology, 416
 - CiscoWorks LMS, 390
 - ClarusIPC Plus⁺ with NNMi, 396
 - ClarusIPC Plus⁺ with NNM iSPI for IP Telephony, 398
 - HPOM agent implementation, 489
 - NA, 426
 - Netcool/OMNIBus, 526
 - nGenius Performance Manager, 453
 - PCM Plus, 440
 - RAMS, MPLS WAN, 441
 - SIM, 444
 - UCMDB, 424
- sweep, ping, 61
- switches
 - defaults, 68
 - defining node groups, 40
 - discovering, 67
 - hierarchies, 41
- synchronization, troubleshooting HPOM, web services implementation, 517

- syslog.log file, 241
- system
 - device support matrix, 31
 - object ID ranges
 - auto-discovery, 64
 - evaluating, 68
 - resources, 83
 - shared file type, HA configuration
 - NNMi, 210
- T**
- tar files, 259
- task flow model, 37
- terms
 - application failover, 180
 - HA, 204
- test environment, 23
- testing
 - communication configuration, 53
 - lag time, 52
 - NNM 6.x/7.x
 - integration, 373
 - traps, 375
- Threshold Alarms category, NNM 6.x/7.x, 370
- timeouts
 - configuring, 46
 - network, 46
 - tuning, 55
 - values, 52
- tips, configuration
 - auto-discovery, 64
 - seeded discovery, 65
- Top-N CPU Utilization portlet, 405
- Top-N Devices by Component Exceptions portlet, 405
- Top-N Memory Utilization portlet, 405
- topology
 - database, 71
 - NNM 6.x/7.x management station, 368
 - scope data, 257
 - summary example URL, 372
- topology filter, BSM topology integration, 421
- traceroute command, 52
- traffic
 - controlling, 52
 - disabling, 49
- trapd.conf file, editing, 368
- traps, testing to NNM 6.x/7.x system, 375

- troubleshooting
 - AlarmPoint integration, 385
 - AlarmPoint Mobile Gateway integration, 387
 - Asset Manager integration, 402
 - BAC My BSM integration, 410
 - BSM Operations Management integration
 - forwarded incidents, 480 to 482
 - BSM topology integration, 418
 - CiscoWorks LMS integration, 392
 - ClarusIPC Plus+ integration
 - with NNMi, 397
 - with NNM iSPI for IP Telephony, 399
 - discovery
 - auto-discovery, 67 to 68
 - HA configuration, 234
 - HPOM integration, agent implementation
 - forwarded incidents, 498 to 501
 - HPOM integration, web services
 - implementation
 - firewall, 518
 - forwarded incidents, 514 to 518
 - message browser, 517
 - synchronization, 517
 - NA integration, 434
 - nGenius Performance Manager integration, 452 to 454
 - NNMi Integration Module for Netcool Software, 532 to 533
 - NNMi northbound interface, 463
 - PCM Plus integration, 440
 - RAMS MPLS WAN integration, 442
 - SIM integration, 447
 - UCMDB integration, 424
- trust store
 - output example, 103
- tuning
 - communication, 55
 - incidents, 96
 - state polling, 82 to 83
- U**
- UCMDB integration
 - benefits, 424
 - changing, 424
 - description, 423
 - disabling, 424
 - documentation, 424
 - parameters, 424
 - supported versions, 424
 - troubleshooting, 424
 - usage, 424
- unconfiguring HA clusters
 - NNMi, 225 to 228

Universal Configuration Management Database. *See*
UCMDB integration

UNIX

disk group

 NNMi, 211

environment variables

 administration, 544

 installation, 24

HA configuration

 files, 239

ipNoLookup.conf command

 excluding addresses from discovery, 340

LicFile.txt file

 configuring primary HA cluster node, 212

 installing HA cluster license, 220

mount point

 NNMi, 211

netmon.cmstr command, 328

nnm8EventForwardDestinations.txt file, 367

nnm.envvars.sh command, 542

nnmtrapd log files, 237

pmd log files, 237

port numbers, 372

remsh command, 202

sendMsg.ovpl command, 375

shared disk directories, 218

ssh command, 202

trapd.conf file, 368

volume group

 NNMi, 211

unmounting shared disk

 NNMi, 227

upgrade tool

 data collection

 archiveMigration.ovpl, 326, 360

 captureLocale.ovpl, 361

 createMigrationDirs.ovpl, 326, 360

 hostnolookup.ovpl, 361

 nnmetmapmigration.ovpl, 363

 nnmmapmigration.ovpl, 357, 361

 nnmmigration.ovpl, 331, 333, 360

 nnmtopodump.ovpl, 359, 361

 ovmapdump.ovpl, 361

 ovmibmigration.ovpl, 361

 ovwdbDump.ovpl, 361

 snmpCapture.ovpl, 361

 trapdConfNodes.ovpl, 361

data import

 nnmconnect.ovpl, 342

 nnmdiscoCfg.ovpl, 340

 nnmincidentCfg.ovpl, 350

 nnmloadnodegroups.ovpl, 357, 359

 nnmloadseeds.ovpl, 336, 342

 nnmmibmigration.ovpl, 350, 363

 nnmtrapdMerge.ovpl, 352, 363

 nnmtrapload.ovpl, 350, 363

 restoreMigration.ovpl, 326, 363

 snmpCapture.ovpl, 328, 359, 363

upgrading from NNM 6.x/7.x

 controlling rate of upgrading, 365

 custom scripts, 359

 discovery, 333 to 343

 events, 349 to 356

 Home Base container views, 358 to 359

 options, 322 to 324

 OVW maps, 356 to 358

 phases, 322

 SNMP information, 327 to 333

 status monitoring, 343 to 349

 suggested path, 297, 299, 307, 321

upgrading NNMi under HA, 230 to 233

upgrading to NNMi 9.00

 configuration differences, 307

 different NNMi management server, 299

 functionality differences, 307, 308

 migrating NNMi Oracle Data, 305

 moving to Red Hat Linux 5.2 or 5.3, 301

 same NNMi management server, 297

 supported upgrades, 295

URL access, SSO, 114

URLs

 NNMi console, 407

 NNM iSPI for Performance report, 407

 not requiring selection, 372

 requiring selection, 373

user interface model, 38

user names in a directory service, 119

UUID, node details using, 373

V

variables, environment

- overview, 541

UNIX

- administration, 544

- installation, 24

- MANPATH, 32

Windows

- administration, 542

- application failover, 180

- installation, 24

- NNMi 8.00, 545

variables, MIB II, 78

verifying

- communication settings, 54

- default settings, 80

- event forwarding, 373

- interface groups, 80

- management IP address, 54

- monitoring configuration, 80 to 81

network

- connectivity, 66

- monitoring configuration, 80

- NNM 6.x/7.x incident configuration, 369

- node groups, 80

- nodes configured for SNMP, 53

- order numbers, 79

- SNMP access for devices, 54

State Poller

- communication settings, 54

- health statistics, 82

Veritas Cluster Server, 202

- HA resource group, 204

- nmharg.ovpl script, 240

version

- information file, 258

- SNMP preferences, 48

version comparisons

customizing

- event monitoring, 318

- discovering networks, 313

- status monitoring, 316

versions, supported

- HA products, 202

- hardware and software, 31

integrated applications

- AlarmPoint, 383

- AlarmPoint Mobile Gateway, 385

- Asset Manager, 402

- BAC My BSM, 404

- BSM Operations Management, 473

- BSM topology, 416

- CiscoWorks LMS, 390

- ClarusIPC Plus⁺ with NNMi, 396

- ClarusIPC Plus⁺ with NNM iSPI for IP

- Telephony, 398

- HPOM agent implementation, 489

- NA, 426

- Netcool/OMNIbus, 526

- nGenius Performance Manager, 453

- PCM Plus, 440

- RAMS, MPLS WAN, 441

- SIM, 444

- UCMDB, 424

viewing NA incident actions, 431

views

- configuring additional NNM 6.x/7.x, 372

- launching dynamic from NNMi, 376

virtual host, HA configuration

netmask

- NNMi, 210

network interface

- NNMi, 210

short name

- NNMi, 210

virtual IP address support, 202

Volume_A.log file, 242

volume group, HA configuration, 211

vxfs shared disk format

- HA, 211

W

web server

- AlarmPoint, 383

- ports, 372

weekly data backups, 260

window, System Information, 82

Windows

- configuring shared disks, 219
- environment variables
 - administration, 542 to 545
 - application failover, 180
 - installation, 24
- HA configuration
 - files, 239
 - scripts, 239
- LicFile.txt file
 - configuring primary HA cluster node, 212
 - installing HA cluster license, 220
- Microsoft Cluster Services, 202
- Microsoft Failover Clustering, 202
- mount point
 - NNMi, 211
- netmon.cmstr command, 328
- netmon.noDiscover command, 340
- nnm8EventForwardDestinations.txt file, 367
- nnm.envvars.bat command, 542 to 545
- nnmhargconfigure.ovpl command, 235
- nnmtrapd log files, 237
- pmd log files, 237
- port numbers, 372
- sendMsg.ovpl commands, 375
- shared disk directories, 218
- shared disk format
 - HA, 211
- trapd.conf file, 368
- virtual hosts
 - NNMi, 210

wizard, NNMi Quick Start Configuration, 23

workspaces

- Configuration
 - configuring NNM 6.x/7.x management station, 371
 - configuring state polling, 78
 - evaluating state polling, 80
- Incidents, 373
- Interface Groups, 75
- Node Groups, 75

X

- XML files, 44, 284
- xnmevents command, NNM 6.x/7.x, 367
- XServer, 368

