HP Network Node Manager i-series ソフトウェア

Windows®、HP-UX、Linux、および Solaris のオペレーティングシステム用

ソフトウェアバージョン：8.10

導入および移行ガイド

ドキュメントのリリース日付：2008 年 11 月
ソフトウェアのリリース日付：2008 年 11 月

ご注意

保証について

HP の製品やサービスに関する保証は、各製品やサービスに付帯する明示保証条項に記載されている通りです。本書は、その他の保証を行うものと解釈されるものではありません。HP は、本書に含まれる技術的または編集上の誤り、欠落についての責任は負いかねます。

ここに記載されている内容は、予告なく変更されることがあります。

権利制限について

機密コンピュータソフトウェアです。これらを所有、使用、または複製するには、HP からの有効なライセンスが必要です。FAR 12.211 および 12.212 に従って、商用コンピュータソフトウェア、コンピュータソフトウェアのドキュメント、および商用アイテムの技術データは、HP の標準商用ライセンス条件に基づいて米国政府にライセンスされています。

著作権について

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

本製品には、Apache Software Foundation (http://www.apache.org/) によって開発されたソフトウェアが含まれています。Portions Copyright © 1999-2003 The Apache Software Foundation. All rights reserved.

本製品には、Institute National de Recherche en Informatique et Automatique (INRIA) によって開発された ASM Bytecode Manipulation Framework ソフトウェアが含まれています。Copyright © 2000-2005 INRIA, France Telecom. All Rights Reserved.

本製品には、Apache Software Foundation (http://www.apache.org/) によって開発された Commons Discovery ソフトウェアが含まれています。Copyright © 2002-2008 The Apache Software Foundation. All Rights Reserved.

本製品には、Netscape JavaScript Browser Detection Library ソフトウェアが含まれています。Copyright © Netscape Communications 1999-2001

本製品には、Apache Software Foundation (http://www.apache.org/) によって開発された Xerces-J xml パーサーソフトウェアが含まれています。Copyright © 1999-2002 The Apache Software Foundation. All rights reserved.

この製品には、Indiana University Extreme! Lab (http://www.extreme.indiana.edu/) で開発されたソフトウェアが含まれています。Xpp-3 Copyright © 2002 Extreme! Lab, Indiana University. All rights reserved.

商標について

DOM4J® は、MetaStuff, Ltd. の登録商標です。

すべての HP 9000 コンピュータ上で動作する HP-UX リリース 10.20 以降および HP-UX リリース 11.00 以降（32 ビットと 64 ビット構成の両方）は、Open Group UNIX 95 による商標の付いた製品です。

Java™ は、Sun Microsystems, Inc. の米国における商標です。

Microsoft® および Windows® は、Microsoft Corporation の米国における登録商標です。

Oracle® は Oracle Corporation およびその関連会社の登録商標です。

UNIX® は The Open Group の登録商標です。

Oracle テクノロジの制限された権限に関する通知

DOD FAR 補足規定に従って供給されたプログラムは、「商用コンピュータソフトウェア」であり、関連文書を含むプログラムの使用、複製、および公開は、該当する Oracle 使用許諾契約に記載されている使用許諾の制限に従うものとします。そうでなければ、連邦調達規則に従って供給されたプログラムは、「制限されたコンピュータソフトウェア」であり、関連文書を含むプログラムの使用、複製、および公開は、FAR 52.227-19、『商用コンピュータソフトウェア - 制限された権限』（1987 年 6 月）に記載されている制限に従うものとします。Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Oracle ライセンスの全文は、NNMi の製品 DVD 上にある license-agreements のディレクトリを参照してください。

使用可能な製品ドキュメント

このガイドに加え、次のドキュメントが NNMi について利用できます。

- *HP Network Node Manager i Software インストールガイド*— 製品メディアおよび NNMi 管理サーバー上でサポートされている各オペレーティングシステムについて入手できます。
- *HP Network Node Manager i Software リリースノート*— 製品メディアおよび NNMi 管理サーバーで入手できます。
- *HP Network Node Manager i Software システムおよびデバイス対応マトリックス* — 製品メディアおよび NNMi 管理サーバーで入手できます。
- *HP Network Node Manager i Software ネットワーク エンジニアリング ツールセットガイド*—製品メディアおよび NNMi 管理サーバーで入手できます。

最近の更新を確認する場合、または最新のドキュメントを使用しているか確認する場合は、以下をご覧ください。

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

このサイトにアクセスするには、HP Passport について登録しサインインする必要があります。HP Passport ID を登録するには、以下にアクセスします。

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

または、HP Passport ログインページの［**新規ユーザー - 登録してください**］リンクをクリックします。

製品のサポートサービスに登録すると、最新版を入手できます。詳細は HP 販売員にお尋ねください。

サポート

次の HP ソフトウェアサポート Web サイトを参照してください。

www.hp.com/managementsoftware/services

この Web サイトには、製品、サービス、および HP Software が提供するサポートの問い合わせ情報および詳細が記載されています。

HP Software Support Online には、お客様の自己解決機能が備わっています。ビジネスを管理するために必要な対話形式のテクニカルサポートツールにアクセスする迅速で効率的な方法が用意されています。お客様は、HP ソフトウェアサポートサイトで、以下の機能を利用できます。

- 関心のあるドキュメントの検索
- サポートケースおよび拡張リクエストの送信および追跡
- ソフトウェアパッチのダウンロード
- サポート契約の管理
- HP サポートの問合せ先の検索
- 利用可能なサービス情報の確認
- ソフトウェアを利用している他のユーザーとの情報交換
- ソフトウェアトレーニング情報の検索および参加登録

大部分のサポートには、HP Passport へのユーザー登録とログインが必要です。また、サポート契約が必要な場合もあります。

アクセスレベルに関する詳細は、次の URL で確認してください。

www.hp.com/managementsoftware/access_level

HP Passport ID のご登録は、次の URL で行ってください。

www.managementsoftware.hp.com/passport-registration.html

目次

このガイドについて 15
ガイドの説明 15
このドキュメントで使用する環境変数 16
改訂履歴 17
NNMi の詳細は、以下を参照してください 18

準備 19

ハードウェアとソフトウェアの要件 21
対応ハードウェアとソフトウェア 21
システム設定 (UNIX) 22
NNMi と HP Performance Insight との共存 23

設定 25

設定の一般概念 27
タスクフローモデル 27
ベストプラクティス：既存の設定を保存 28
ベストプラクティス：作成者属性を使用する 28
ユーザーインタフェースモデル 28
順序 28
ノードグループ 29
グループの重複 30
ノードグループ 31
階層 / コンテインメント 31
デバイスフィルタ 32
追加フィルタ 32
追加ノード 32
インタフェースグループ 33
ノード / インタフェース / アドレス階層 33
すべてを停止して再度やり直す 33

NNMi 通信 35
通信の概念 35
ネットワーク待ち時間とタイムアウト 36
アクセス資格認定とコミュニティ文字列 36
SNMP バージョンの優先 36
設定レベル 37
優先される管理アドレス 38

ポーリングプロトコル 38
通信の計画作成 39
意味のあるデフォルト 39
通信設定領域 39
特定のノードの設定 40
再試行とタイムアウトの値 40
アクティブなプロトコル 40
複数のコミュニティ文字列またはアクセス資格認定 41
通信の設定 41
通信の評価 41
すべてのノードが SNMP 用に設定されましたか？ 42
デバイスについて SNMP アクセスは現在利用できますか？ 42
NNMi は正しい通信設定を使っていますか？ 42
State Poller 設定は通信設定と一致していますか？ 43
管理 IP アドレスは正しいですか？ 43
通信の調整 43

NNMi 検出 45
検出の概念 45
NNMi はデバイスプロファイルから属性を導き出す 46
検出の計画 47
基本的な検出方法を選択する 47
リストベース検出 47
ルールベース検出 47
自動検出ルール 48
自動検出ルールの順序 48
デバイスを検出から除外 48
Ping スィープ 49
自動検出ルールの検出シード 49
自動検出ルールのベストプラクティス 49
例 49
ノード名の解決 50
サブネット接続ルール 51
検出シード 51
再検出の間隔 51
オブジェクトを検出しない 52
検出の設定 52
自動検出ルールを設定する場合のヒント 53
シードを設定する場合のヒント 53
検出の評価 54
初期検出の進行状況をたどる 54
すべてのシードが検出されているか？ 54
すべてのノードには有効なデバイスプロファイルがあるか？ 55
すべてのノードが正しく検出されたか？ 55
自動検出ルール 55
IP アドレス範囲 55
システムオブジェクト ID の範囲 56
すべての接続と VLAN は正しいか？ 56

デバイスを再検出する 56
検出の調整. 56

NNMi 状態ポーリング 57
状態ポーリングの概念. 57
状態ポーリングの計画を作成 58
ポーリングチェックリスト 58
何をモニタリングできるでしょうか？ 59
モニタリングされないノードへのインタフェース. 60
モニタリングの停止 61
グループの計画作成. 61
インタフェースグループ. 62
ノードグループ. 62
ポーリング間隔の計画作成 63
どのデータを収集するかの決定. 64
状態ポーリングの設定. 64
インタフェースグループとノードグループの設定 64
インタフェースモニタリングの設定 65
ノードモニタリングの設定 65
デフォルト設定の確認 66
状態ポーリングの評価. 66
ネットワークモニタリングの設定を確認します 66
インタフェースまたはノードは正しいグループのメンバーでしょうか？. 66
どの設定が適用されていますか？ 67
どのデータが収集されていますか？. 67
ステータスポーリングのパフォーマンスの評価 67
State Poller は最新の状態に付いていっていますか？. 68
状態ポーリングの調整. 68

管理 71

https の有効化（対象：NNMi). 73
パブリックキー証明書の入手とインストール. 73
自己署名証明書 73
第三者証明書 74
server.xml ファイルの更新. 74
新しい接続情報をユーザーに通信. 76

高可用性クラスタに HP NNM i-series Software を設定する 77
サポート対象の HA 製品. 78
HA 用に NNMi を設定するための前提条件. 78
HA の概念. 79
HA 用語集 80
NNMi HA クラスタのシナリオ. 80
マンページ 83
HA の設定. 84

HA 用の NNMi の設定 84
NNMi HA 設定情報 84
プライマリクラスタノードでの NNMi の設定. 86
セカンダリクラスタノードでの NNMi の設定. 88
HA 用の iSPI の設定 89
iSPI HA 設定情報. 89
NNM iSPI for Performance をアドオン iSPI として設定. 89
アドオン iSPI の設定（他のすべての iSPI). 90
専用サーバーで HA 用に NNM iSPI for Performance を設定する 91
NNM iSPI for Performance HA 設定情報. 91
プライマリクラスタノードでの NNM iSPI for Performance の設定 93
セカンダリクラスタノードに NNM iSPI for Performance を設定する 94
Oracle 環境で、HA 用に NNMi を設定する. 95
Oracle を使っている NNMi の HA 設定情報. 95
Oracle を使っている NNMi に HA を設定する. 95
共有 NNMi データ 96
NNMi の共有ディスク内のデータ. 96
ファイルレプリケーションの設定 97
共有ディスクの設定. 97
Windows 2003 での共有ディスク設定についての注記. 98
HA クラスタ内の NNMi のライセンス契約. 98
HA 設定のメンテナンス 99
メンテナンスモード. 99
HA リソースグループをメンテナンスモードにする 99
HA リソースグループのメンテナンスモードを解除する. 99
HA クラスタ内の NNMi のメンテナンス 99
NNMi の起動と停止 99
クラスタ環境で NNMi のホスト名や IP アドレスを変更する. 99
フェイルオーバーを行わせないように NNMi を停止する. 102
メンテナンス後に NNMi を再起動する 102
NNMi HA クラスタ内のアドオン iSPI のメンテナンス. 102
HA クラスタ内の専用サーバー上の NNM iSPI for Performance のメンテナンス 102
NNM iSPI for Performance の起動と停止 102
クラスタ環境で NNM iSPI for Performance システムを変更する 103
フェイルオーバーを行わせないように NNM iSPI for Performance を停止する 103
メンテナンス後に NNM iSPI for Performance を再起動する. 103
HA の設定解除 104
HA クラスタ内の NNMi の設定を解除する 104
HA クラスタ内の NNM iSPI for Performance の設定を解除する 107
HA 下の NNMi を NNMi 8.0x から NNMi 8.10 にアップグレードする. 110
HA 設定のトラブルシューティング 113
一般的な HA のトラブルシューティング 113
エラー：Wrong Number of Arguments（引き数の数が間違っています）. 113
製品スタートアップのタイムアウト (Solaris) 113
アクティブなクラスタノードのログファイルが更新されない. 113
HA リソースグループが特定のクラスタノードでは起動できない 114

NNMi 固有の HA のトラブルシューティング 115
すべてのクラスタノードの設定解除後に、HA 用の NNMi を再び有効にする 115
NNMi を HA 下で正常に起動できない 115
HA の設定後、nmsdbmgr を起動できない 116
HA の設定後、pmd を起動できない 116
ディスクフェイルオーバーが行われない 116
フェイルオーバー後にセカンダリノードで共有ディスクファイルが見つからない 117
iSPI 固有の HA のトラブルシューティング 117
iSPI を HA 下で正常に起動できない 117
HA 設定リファレンス 118
NNMi HA 設定ファイル 118
NNM iSPI for Performance HA 設定ファイル 118
NNMi に付属している HA 設定スクリプト 118
NNMi HA 設定のログファイル 120
NNM iSPI for Performance HA ログファイル 121

NNMi のバックアップおよびリストアツール 123
NNMi データバックアップについて 123
NNMi バックアップおよびリストアスクリプトの制限 124
バックアップシステムとリストアシステムは、同じ設定でなければならない 124
外部データベースはサポートされない 124
NNMi データのバックアップ 125
NNMi データのリストア 127
組み込みデータベースのみをバックアップおよびリストアする 128

NNMi 管理サーバーの変更 131
NNMi 設定移動の準備のベストプラクティス 131
NNMi 設定の移動 132
NNMi 管理サーバーの IP アドレスの変更 132

NNM 6.x/7.x からの移行 133

製品の比較 135
ネットワーク検出 136
検出の重要概念 138
ステータスモニタリング 139
ステータスモニタリングの重要概念 141
イベントモニタリングのカスタマイズ 142
イベントモニタリングの重要概念 144

NNM 6.x/7.x からの移行 145
移行オプション 146
新たな始まり 146
フェーズのある移行 146
フェーズ 1 : SNMP 情報の移行 148
SNMP アクセスの設定 148
名前解決の制限 151
デバイスプロファイルのカスタマイズ 151

フェーズ 2：検出の移行 152
検出のスケジュール 152
自分の検出方法の選択 154
自動検出ルールの設定 155
スパイラル検出の設定 155
シード検出のためにシードを NNMi に追加します 159
接続のカスタマイズ 160
フェーズ 3：ステータスモニタリングの移行 161
ポーリング間隔の設定 161
ポーリングプロトコルの選択 162
危険域にあるノードの設定 165
ステータスポーリングからオブジェクトを除外します 166
フェーズ 4：イベント設定とイベント削減の移行 167
デバイスからのトラップの表示 167
NNMi で生成された管理イベント表示のカスタマイズ 169
トラップのブロック / 無視 / 無効化 169
自動処理の設定 170
追加（手動）処理の設定 170
イベント相関処理：イベントの繰り返し 170
イベント相関処理：レート計算 171
イベント相関処理：Pairwise のキャンセル 172
イベント相関処理：スケジュールされたメンテナンス 172

NNM 6.x または NNM 7.x と NNMi 8.10 との統合 173

イベント転送の設定 174
ステップ 1：NNM 6.x/7.x を、NNMi 8.10 管理サーバーにイベントを転送するように設定する 174
推奨およびサポートされる手順：［イベント設定］ウィンドウを使用する 174
任意：転送先リストファイル 175
別の手順：trapd.conf を手順で編集する 176
ステップ 2：（任意）ノードレベルのフィルタリングを使用してイベント数をさらに削減する 176
ステップ 3：NNM 6.x/7.x 管理ステーションを NNMi 8.10 トポロジに追加する 176
ステップ 4：（任意）管理ステーション設定を保存する 176
ステップ 5：NNM 6.x/7.x インシデント設定を NNMi 8.10 コンソールで確認する 177
カテゴリのマッピング 177
リモートビュー起動の設定 178
ステップ 1：Java Run-Time Environment (JRE) のインストール 178
ステップ 2：NNMi 8.10 で NNM 6.x/7.x 管理ステーションエンティティを作成する 178
ステップ 3：（任意）追加の NNM 6.x/7.x ビューを設定する 179
選択を必要としない URL 180
選択を必要とする URL 180
統合をテストする 180
テスト 1：イベント転送を確認する 181
ノード / インタフェースの停止 / 稼働イベントを発生させる 181
sendMsg.ovpl 182
NNM 6.x/7.x システムへのトラップをテストする 182
テスト 2：NNM 6.x/7.x 動的ビューを NNMi 8.10 から起動する 183
イベント転送のトラブルシューティング 183

統合とプラグイン 185

HP Operations Manager 187
HP NNMi–HPOM 統合 187
値 188
サポートされるバージョン 189
ドキュメント 189
HP NNMi–HPOM 統合の有効化 189
Windows 用 HPOM 190
UNIX 用 HPOM 191
HP NNMi–HPOM 統合の使用法 193
NNMi–HPOM 統合設定の変更 194
HP NNMi–HPOM 統合の無効化 194
すべての HPOM 管理サーバーについて 194
1 つの HPOM 管理サーバーについて 195
HP NNMi–HPOM 統合のトラブルシューティング 195
HPOM は転送されたインシデントを受信しない 195
転送されたインシデントの中に HPOM が受信しないものがある 197
NNMi インシデント情報が HPOM メッセージブラウザでは入手できない 197
NNMi と HPOM が同期されない 197
統合がファイアウォールを経由して動作しない 197
HP NNMi–HPOM Integration Configuration フォームリファレンス 198
NNMi 管理サーバー接続 198
HPOM Management Server Connection（HPOM 管理サーバー接続） 199
統合動作 200
Incident Filter（インシデントフィルタ） 200
インシデントフィルタの例 202
インシデントフィルタの制限 203
HP NNMi–HPOM 統合ログファイルリファレンス 204
ログファイル 204
nnm-om ログファイル内の詳細メッセージを有効にする 204
jboss-log4j ログファイルで統合通信メッセージを有効にする 205
ログファイルメッセージの例 205
Good Start up Trace（良好なトレースの起動） 206
UNIX 用に HPOM に正常に転送された NNMi インシデント 207
UNIX 用に HPOM で正常に閉じられた NNMi インシデント 207
正しくない HPOM ログイン設定 207
正しくない NNMi ログイン設定 208
Windows 用の HPOM からの肯定応答エラー 209
UNIX 用の HPOM からの肯定応答エラー 210
ノードが見つからないというエラー 212

HP Universal Configuration Management Database
（HP ユニバーサル設定管理データベース） 213
HP NNMi–HP UCMDB 統合 213
値 214
サポートされるバージョン 214

ドキュメント 214
HP NNMi–HP UCMDB 統合の有効化 214
HP NNMi–HP UCMDB 統合の使用法 215
影響対象 CI の表示 215
UCMDB CI の表示 216
HP NNMi–HP UCMDB 統合設定の変更 216
HP NNMi–HP UCMDB 統合の無効化 217
HP NNMi–HP UCMDB 統合のトラブルシューティング 217
HP NNMi–HP UCMDB Integration Configuration フォームリファレンス 217
NNMi 管理サーバー接続 218
UCMDB サーバー接続 219
統合動作 219

AlarmPoint 221
HP NNMi–AlarmPoint 統合 221
価値 222
サポートされるバージョン 222
ドキュメント 222
HP NNMi–AlarmPoint 統合を有効にする 223
HP NNMi–AlarmPoint 統合の使用法 223
HP NNMi–AlarmPoint 統合を無効にする 224
HP NNMi–AlarmPoint 統合のトラブルシューティング 224

nGenius Performance Manager 225
HP NNMi–nGenius Performance Manager 統合 226
値 226
サポートされるバージョン 226
ドキュメント 227
HP NNMi–nGenius Performance Manager 統合の有効化 227
HP NNMi–nGenius Performance Manager 統合の使用法 227
HP NNMi–nGenius Performance Manager 統合の無効化 228
HP NNMi–nGenius Performance Manager 統合のトラブルシューティング 228

NNM iSPI for Performance 229
NNM iSPI for Performance とは何か 229
値 230
サポートされるバージョン 230
ドキュメント 230
NNM iSPI for Performance の有効化 230
HA で実行されるよう NNM iSPI for Performance を設定 231
NNM iSPI for Performance の使用法 231
NNM iSPI for Performance のバックアップを規定 231
NNM iSPI for Performance のリセット 232
NNM iSPI for Performance を異なるコンピュータに移動 232
NNM iSPI for Performance の無効化 232
NNM iSPI for Performance のトラブルシューティング 232

追加情報 233

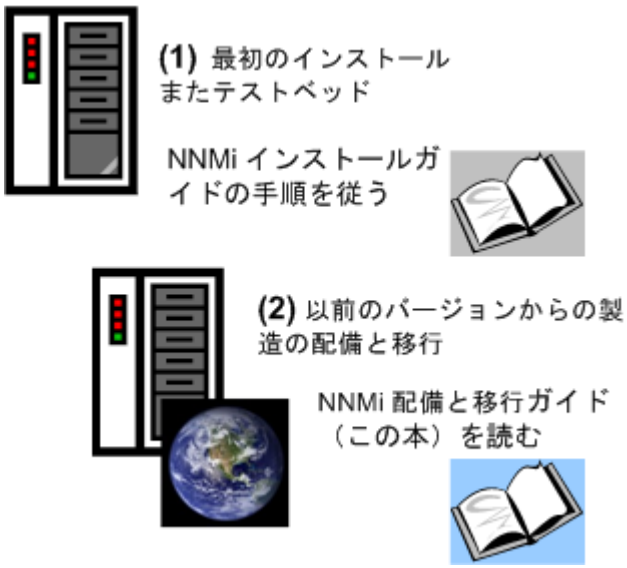
NNMi 環境変数 235
このドキュメントで使用する環境変数 235
他の使用可能な環境変数 236
NNMi 8.00 からの Windows パスおよび環境変数 238

Causal Engine と NNMi インシデント 241
因果関係解析 - 高度な考察 241
Causal Engine の概念 242
ステータスの概念 242
エピソードとは 243
NNMi は何を解析するのか？ 243
失敗のシナリオは何ですか？ 246
SNMP エージェントが SNMP クエリーに応答しない 247
SNMP エージェントが SNMP クエリーに応答している 248
IPv4 アドレスが ICMP に応答しない 249
ICMP への IPv4Address 応答 250
インタフェースを操作できない 251
インタフェースは操作可能である 252
インタフェースを管理できない 253
インタフェースを管理できる 254
接続を操作できない 255
接続を操作できる 256
直接接続しているノードが停止している 257
直接接続されたノードは稼働している 258
間接接続されたノードは停止している 259
間接接続されたノードは稼働している 260
直接接続されたノードが停止しており、シャドウを作成する 261
直接接続されたノードが稼働しており、シャドウを除去している 262
重要ノードが到達不可能である 263
重要ノードが到達可能である 263
ノードまたは接続が停止している 264
ノードまたは接続が稼働している 264
アイランドグループが停止している 265
アイランドグループが稼働している 266
ネットワーク設定の変更 267
ノード更新中 267
インタフェースが接続に加入および離脱する 267
デバイスがトラップを発生した場合 267
NNMi 管理設定の変更 268

用語集 269

索引 277

このガイドについて



この章では以下について説明します。

- ガイドの説明
- このドキュメントで使用する環境変数
- 改訂履歴
- NNMi の詳細は、以下を参照してください

ガイドの説明

このガイドには、NNMi や NNMi Advanced など、HP Network Node Manager i Software を導入するための情報およびベストプラクティスが記載されています。対象読者は、熟練したシステム管理者、ネットワークエンジニア、または大規模システムのネットワーク導入および管理に経験のある HP サポートエンジニアです。

このガイドでは、制限のある環境（またはテスト環境）に NNMi をインストール済みであること、クイックスタート設定ウィザードを使用したコニュミティ文字列の設定、ネットワークノードの制限範囲の検出設定、初期管理者アカウントの作成のような、設定作業の開始に慣れていることを仮定しています。これら作業の詳細は、*NNMi インストールガイド*を参照してください（使用可能な製品ドキュメントページ 3 を参照）。

新しい情報が入手可能になると、製品リリースの間に、HP はこの導入および移行ガイドを更新します。ドキュメントの更新バージョン取得の詳細は、使用可能な製品ドキュメントページ 3 を参照してください。

このドキュメントで使用する環境変数

このドキュメントでは、ファイルとディレクトリの場所を参照するために、次の 2 つの NNMi 環境変数を使用します。デフォルト値は次のとおりです。実際の値は、NNMi のインストールの間に行った選択で決まります。

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

Windows システムでは、これらの変数は NNMi インストールプロセスによって作成されるので、いつでも使用できます。

最初に NNMi 8.00 をインストールした場合、ご使用のシステムでは、NNMi 8.00 からの Windows パスおよび環境変数ページ 238 で説明しているようにこれらの環境変数で異なる値を使用しています。

- *UNIX®* の場合 :
  - `$NnmInstallDir`: `/opt/OV`
  - `$NnmDataDir`: `/var/opt/OV`

UNIX システムでは、これらの環境変数を使用する場合は手動で作成する必要があります。

入手できる他の NNMi 環境変数の詳細は、NNMi 環境変数ページ 235 を参照してください。

改訂履歴

次の表に、このドキュメントの新規リリースごとの主要な変更をリストします。

ドキュメントリリース日	主要な変更の説明
NNMi バージョン 8.00：	
2007 年 11 月	英語初版。
NNMi バージョン 8.01：	
2008 年 12 月	• NNMi 環境変数で Windows パスを更新。 • NNMi の https の有効化で https 情報を更新。 • ネットワークの検出への追加： — ベストプラクティス：既存の設定を保存 — すべてを停止して再度やり直す • ネットワークノードのモニタリングへの追加： — ベストプラクティス：既存の設定を保存 — モニタリングされないノードでの重要なインタフェースのモニタリング • NNMi を製品環境に移動への追加： — NNMi 設定移動の準備のベストプラクティス • 追加された NNMi 8.00 から NNMi 8.01 へのアップグレード • 追加された NNM iSPI 考慮事項
NNMi バージョン 8.02：	
2008 年 6 月	• NNMi バックアップおよびリストアツールへの追加： — NNMi バックアップおよびリストアスクリプトの制限 • 追加された高可用性クラスタへの NNMi の設定
NNMi バージョン 8.03：	
2008 年 7 月	• 追加された NNM 6.x/7.x からの移行 • 追加された HP オペレーションマネージャとの統合
NNMi バージョン 8.10：	
2008 年 11 月	• 英語第 2 版。完全に更新。 • 日本語初版。

NNMi の詳細は、以下を参照してください

NNMi 製品の完全な情報を入手するには、このガイドと他の NNMi ドキュメントを一緒に使用してください。次の表に、現在までのすべての NNMi ドキュメントを示します。ガイドとホワイトペーパーの両方を含みます。

情報はすべて **http://h20230.www2.hp.com/selfsolve/manuals からダウンロードできます。**詳細は、使用可能な製品ドキュメントページ 3 を参照してください。

何を行うか？	さらに情報を見つける場所
NNMi または NNMi Advanced をインストール（初回）。	『*NNMi インストール ガイド*』のダウンロード。このガイドには、製品をインストールおよびアンインストールする基本手順、および NNMi クイックスタート設定ウィザードを使用して初期設定を行う方法が記載してあります。 • *HP Network Node Manager i-series Software『Windows オペレーティングシステムのインストールガイド』* • *HP Network Node Manager i-series Software『HP-UX オペレーティングシステムのインストールガイド』* • *HP Network Node Manager i-series Software『Linux オペレーティングシステムのインストールガイド』* • *HP Network Node Manager i-series Software『Solaris オペレーティングシステムのインストールガイド』*
ネットワーク導入の計画。システム要件へのリンクも含まれます。	このガイドの準備ページ 19 を参照してください。
製品環境向けに NNMi を設定します。	このガイドの設定ページ 25 を参照してください。
NNMi の管理。高可用性 (HA)、またはバックアップを含みます。	このガイドの管理ページ 71 を参照してください。
Network Node Manager の前バージョンから NNMi に移行します。	このガイドの NNM 6.x/7.x からの移行ページ 133 を参照してください。
NNMi iSPIs など、NNMi と統合される製品について詳細を学んでください。	このガイドの統合とプラグインページ 185を参照してください。
HP NNMi iSPI Network Engineering Toolset (iNET) をインストールし、これのコンポーネントについて学んでください。	『*HP Network Node Manager i-series Software iSPI Network Engineering Toolset Planning and Installation Guide*』をダウンロードします。
NNM Developer's Toolkit (SDK) のドキュメントを入手します。	『*NNMi Developer's Toolkit Guide*』をダウンロードします。

準備

この項では以下の章について説明します。

- ハードウェアとソフトウェアの要件

ハードウェアとソフトウェアの要件

この章では以下について説明します。

- 対応ハードウェアとソフトウェア
- システム設定 (UNIX)
- NNMi と HP Performance Insight との共存

対応ハードウェアとソフトウェア

NNMi をインストールする前に、表 1 に説明されている NNMi のハードウェアとソフトウェアの要件に関する情報を読んでください。

上記の最新版のドキュメントは、以下から入手してください。
http://h20230.www2.hp.com/selfsolve/manuals

表 1　ソフトウェアおよびハードウェアのプレインストールのチェックリスト

チェック欄（はい / いいえ）	確認していただくドキュメント
	NNMi インストール ガイド • ファイル名 = `install-guide_ja.pdf` • **Windows** メディア = DVD メインドライブ (root) • **UNIX** メディア = ルートディレクトリ • **NNMi** コンソール = **［ヘルプ］** > **［NNMi ドキュメントライブラリ］** > **［インストールガイド］**
	HP NNMi ソフトウェアリリースノート • ファイル名 = `releasenotes_ja.html` • **Windows** メディア = DVD メインドライブ (root) • **UNIX** メディア = ルートディレクトリ • **NNMi** コンソール = **［ヘルプ］** > **［NNMi ドキュメントライブラリ］** > **［リリースノート］**
	HP NNMi ソフトウェアシステムおよびデバイス対応マトリックス • ファイル名 = `supportmatrix_ja.html` • **Windows** メディア = DVD メインドライブ (root) • **UNIX** メディア = ルートディレクトリ • **NNMi コンソール** = リリースノートからリンクしている

➤ 新しい情報が入手可能になった時点で、HP は *HP NNMi ソフトウェアシステムおよびデバイス対応マトリックス*を更新しています。NNMi を導入する前に、以下の Web サイトで、お持ちのバージョンのソフトウェアに関する最新の NNMi 対応マトリックスをチェックしてください。

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

（この Web サイトにアクセスするには、HP Passport の ID が必要です。）

➤ スマートプラグイン (iSPIs) をインストールする場合は、NNMi 導入時に、これらの製品のシステム要件を組み入れてください。

システム設定 (UNIX)

NNMi 管理サーバーに NNMi のマンページを表示できない場合は、`MANPATH` 変数に `/opt/OV/man` の場所が含まれていることを確認します。含まれていない場合は、`/opt/OV/man` の場所を `MANPATH` 変数に追加します。

NNMi と HP Performance Insight との共存

HP Performance Insight と同じサーバーに NNMi をインストールする場合は、次の手順に従って、インストールシーケンスとポート矛盾の問題を回避してください。

1 HP Performance Insight を最初にインストールします。

手順 1 と 手順 2 を完了してから、NNMi をインストールします。

2 HP Performance Insight の全プロセスを停止します。

3 NNMi をインストールします。特定の指示については『*NNMi インストール ガイド*』を参照してください。

4 次のコマンドで NNMi の全プロセスを停止します。

```
ovstop -c
```

5 ポート矛盾を解決するには `nnm.port.properties` ファイルを変更します。このファイルは次のディレクトリにあります。

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

6 HP Performance Insight プロセスを開始します。

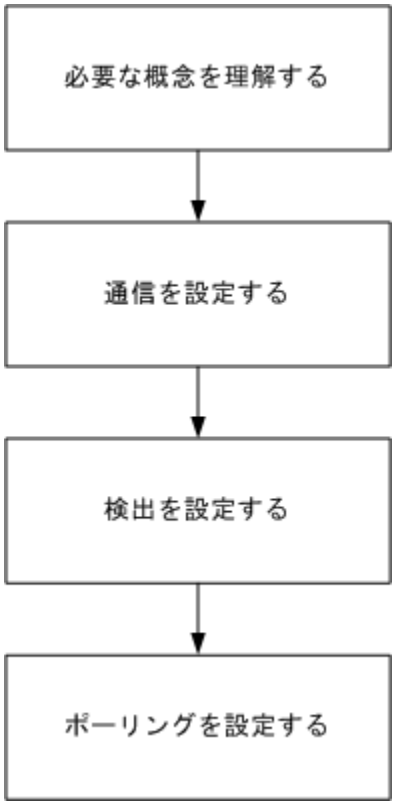
7 次のコマンドで NNMi の全プロセスを開始します。

```
ovstart -c
```

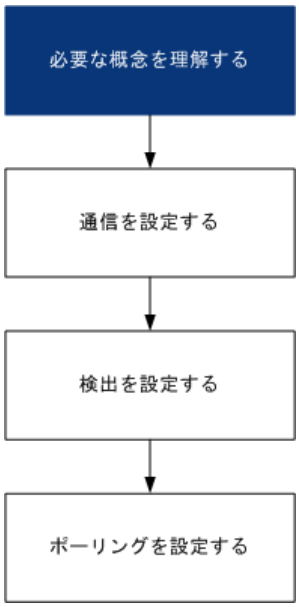
設定

この項では以下の章について説明します。

- 設定の一般概念
- NNMi 通信
- NNMi 検出
- NNMi 状態ポーリング



設定の一般概念



この章では概念の概論を説明しています。詳細については、このガイドの後のほうで説明しています。この章では、すべての HP Network Node Manager i Software 設定領域に適用されるベストプラクティスについても記載しています。

この章では、以下の内容を記載しています。

- タスクフローモデル
- ベストプラクティス：既存の設定を保存
- ベストプラクティス：作成者属性を使用する
- ユーザーインタフェースモデル
- 順序
- ノードグループ
- ノード / インタフェース / アドレス階層
- すべてを停止して再度やり直す

タスクフローモデル

このガイドの設定の各章では、以下のタスクフローに役立つ情報を記載しています。

1. **概念** — 設定領域の概略を理解できます。このガイドの情報は、NNMi ヘルプの情報を補足しています。
2. **計画** — 設定にどう取り組むかを決定します。これは、会社のネットワーク管理のマニュアル化を開始または更新するよい機会です。
3. **設定** —NNMi コンソール、設定ファイル、コマンドラインインタフェースを組み合わせて使用して、設定を NNMi に入力します。具体的な手順については、NNMi ヘルプを参照してください。
4. **評価** —NNMi コンソールで、設定の結果を検査します。設定を最適なものにするために、必要に応じて調節します。
5. **調整** — 任意。設定を調整して、NNMi のパフォーマンスを向上します。

ベストプラクティス：既存の設定を保存

大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。設定を変更した結果が気に入らなくても、保存した設定に簡単に戻すことができます。

`nnmconfigexport.ovpl` コマンドを使用して、現在の設定を保存します。保存した設定を復旧するには、`nnmconfigimport.ovpl` コマンドを使用します。

これらのコマンドを使用する方法については、該当する参照ページか UNIX のマンページを参照してください。

ベストプラクティス：作成者属性を使用する

以下の設定フォームには、**作成者**属性が含まれています。

- デバイスプロファイル
- インシデントの設定
- URL アクション

これらのフォーム上で設定を作成や変更する際は、作成者の組織がわかる値に［**作成者**］属性を設定してください。NNMi 設定をエクスポートするときに、作成者値を指定して作成者の組織がカスタマイズした項目のみを引き出すことができます。

ユーザーインタフェースモデル

NNMi コンソールでは、データベース更新についてトランザクションアプローチを使用します。NNMi コンソールのフォームで行った変更は、フォームを保存して閉じる操作が NNMi コンソールまで行われないと有効になりません。

➤ ［**検出シード**］フォームは、例外です。このフォームは便宜上［**検出の設定**］フォーム上にありますが、他の検出設定からは切り離されています。このため、［**検出の設定**］フォームを保存して閉じて自動検出*ルール*を実装した後で、これらの自動検出ルールに検出シードを設定する必要があります。

順序

いくつかの NNMi コンソール設定フォームには、設定を適用する優先順位を設定する**順序**属性が含まれています。ある設定領域で、NNMi は設定内容に対して各項目を、順序番号が最も小さい（低い）ものから大きいものへの順に、NNMi が一致を見つけるまで評価し続けます。一致が見つかった時点で、NNMi は一致する設定の情報を使用し、これ以上一致を探すのをやめます。（通信設定は例外です。NNMi は、通信設定を完了するために情報を探し続けます。）

順序属性は、NNMi の設定で重要な役割を果たします。予想外の検出結果やステータス結果に遭遇した場合は、その領域の設定の順序を確認してください。

以下の設定では、**順序**属性を使用します。

- **［通信の設定］**フォームの**［領域］**
- **［検出の設定］**フォームの**［自動検出ルール］**
- **［モニタリングの設定］**フォームの**［インタフェース設定］**
- **［モニタリングの設定］**フォームの**［ノードの設定］**
- (NNMi Advanced) **ノードグループごとの設定**：**［インシデントの設定］**フォームのトラップまたはイベント設定の一部です

順序番号は次の箇所でも使用されますが、その意味は異なります。

- **［URL アクション］**フォームの順序は、**［アクション］**メニューの項目の順序を設定します。
- **［ノードグループマップの設定］**フォームの順序は、**［トポロジマップ］**ワークスペースの項目の順序を設定します。

順序属性が所定の設定領域にどう影響するかの具体的内容については、NNMi ヘルプでその領域を参照してください。

ベストプラクティス　各設定領域で、小さい順序番号は最も限定的な設定に適用し、大きな順序番号は限定度の最も低い設定に適用します。

ベストプラクティス　各設定領域で、すべての順序番号を一意にしてください。初期設定時は、通常の間隔の順序番号を使用して、将来設定を変更できるような柔軟性を確保しておいてください。たとえば、1 番目から 3 番目の設定には 100、200、300 の順序番号を付けます。

ノードグループ

NNMi の基本的なフィルタリング手法では、ノードまたはインタフェースをグループ化し、それから設定をグループに適用するか、可視化をグループ別にフィルタリングします。ノードグループは、以下のいずれかまたはすべての目的に使用できます。

- モニタリング設定
- テーブルフィルタリング
- マップビューのカスタマイズ

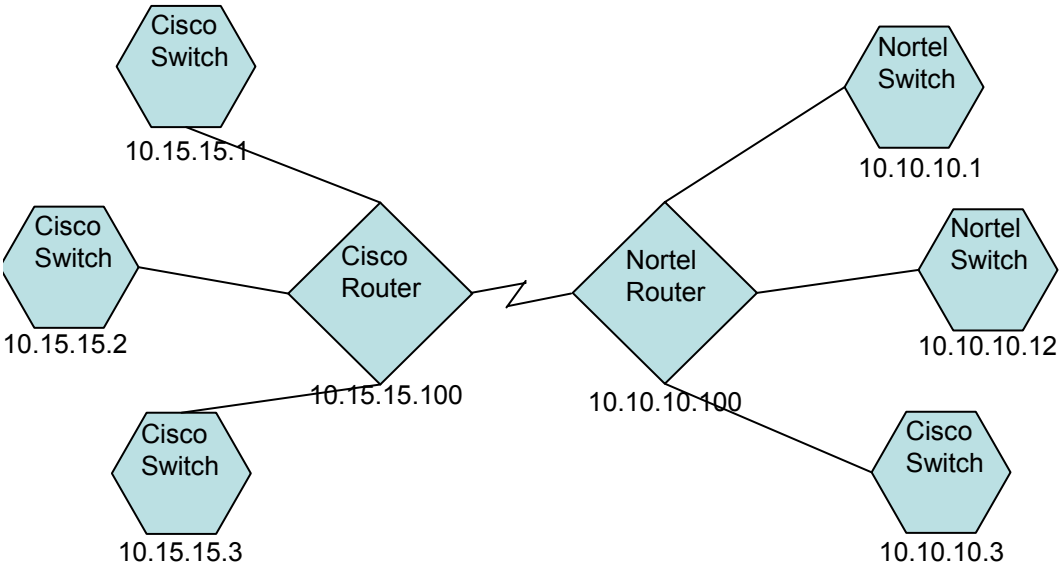
インタフェースグループは、以下のいずれかまたは両方の目的に使用できます。

- モニタリング設定
- テーブルフィルタリング

あらゆるフィルタリング可能な属性に基づいてノードグループの階層を作成して、マップビューのドリルダウンやモニタリング設定の継承を管理できます。

グループの重複

グループ定義をどのように使用するかにかかわらず、最初のステップでは、どのノードやインタフェースをグループのメンバーにするかを定義します。様々な目的でグループが作成されるため、各々の対象が複数のグループに含まれる可能性があります。次の例を考えてみます。



- モニタリングを目的とした場合、ベンダーや場所を問わずすべてのスイッチに 3 分間のポーリング間隔を設定するのがよいでしょう。この場合は、デバイスカテゴリフィルタを使用します。
- 保守を目的とした場合は、すべての Cisco スイッチを 1 つのグループにして、IOS アップグレードではこのグループをまとめてサービス停止にできるようにするのがよいでしょう。この場合は、ベンダーフィルタを使用します。
- 可視化の場合は、10.10.*.* サイト上のすべてのデバイスを、ステータスを反映したコンテナにグループ化するのがよいでしょう。この場合は、IP アドレスフィルタを使用します。

IP アドレスが 10.10.10.3 の Cisco スイッチはこの 3 つのグループすべてに適しています。

設定や表示に便利なようにグループセットを豊富にするのもよいですが、使用されることのない必要以上のエントリを一覧に詰め込みすぎることのないよう、バランスをとってください。

ノードグループ

ノードグループのメンバーシップは、検出した各ノードを設定した各々のグループと比較することにより評価されます。次の 4 種類の情報を使用して、適したノードをフィルタリングできます。

- 明示的に指定する追加ノード

以下の情報で OR を適用

- 子グループを含むこと

以下の情報で OR を適用

  — システムオブジェクト ID から導き出される機能に基づいたデバイスフィルタ

  以下の情報で AND を適用

  — 他の属性に基づいた追加フィルタ

追加するノードは、グループのメンバーになります。子グループのメンバーは、このグループのメンバーとしての資格を得ます。他のノードの場合は、**[デバイスフィルタ]** タブと **[追加フィルタ]** タブを AND 適用した条件と一致する必要があります。**[追加フィルタ]** タブ内では、AND、OR、および括弧評価の複合論理を作成できます。

階層 / コンテインメント

単純で再利用可能な原子グループを作成し、これらをモニタリングや可視化のために階層的に組み合わせることができます。階層的なノードのコンテナを使用することにより、障害時にオブジェクトの場所やタイプに関する手がかりが得られるので、マップビューが大きく向上します。NNMi により、グループの定義とそのドリルダウン順序の徹底管理が可能になります。

単純で再利用可能な原子グループを最初に作成し、その後にこれらを増築するときの子グループとして指定します。また、最初に一番大きな親グループを指定し、それから子グループを作成していくこともできます。

たとえば、ネットワークが Cisco スイッチ、Cisco ルーター、Nortel スイッチ、Nortel ルーターで構成されているとします。Cisco デバイスの親グループとすべてのスイッチの親グループを作成できます。親を作成してその子を指定するときに階層が指定されるので、Cisco スイッチのようなそれぞれの子グループには複数の親ができる可能性があります。

階層は、以下の状況で使用すると効果的です。

- モニタリングニーズが類似したノードのタイプ
- ノードの地理的な配置
- まとめてサービス停止にするノードのタイプ
- オペレータの職務別のノードのグループ

マップビューおよびテーブルビューでグループを使用すると、伝播された（設定可能な）グループのステータスが表示されます。

➤ グループ定義を使用してモニタリング設定を指定する際に階層は設定の順序を示すので*はない*ことを留意してください。小さい順序番号の設定は、ノードに適用されます。順序番号を注意深く増分することで、設定の継承概念を真似ることができます。

設定インタフェースでは、循環階層の定義が自動的に防御されます。

デバイスフィルタ

検出中、NNMi は直接情報を SNMP クエリーで収集し、そこから他の情報を、デバイスプロファイルを通じて導き出します。（詳細については、NNMi はデバイスプロファイルから属性を導き出すページの 46 を参照してください。）システムオブジェクト ID を収集することにより、NNMi は正しいデバイスプロファイルを通じて索引化して、次の情報を導き出します。

- ベンダー
- デバイスカテゴリ
- カテゴリ内のデバイスファミリー

導出されたこれらの値は、デバイスプロファイルそのものとともに、フィルタとして使用できます。

たとえば、特定のベンダー製のすべての対象物を、デバイスタイプやファミリーに関係なくグループ化できます。また、ある種類のデバイス（たとえばルーター）をすべて、ベンダーを問わずにまとめることができます。

追加フィルタ

追加のフィルタエディタを使用すると、以下のようなフィールドに一致するカスタム論理を作成できます。

- hostname（ホスト名）
- mgmtIPAddress（管理アドレス）
- hostedIPAddress（アドレス）
- sysName（システム名）
- sysLocation（システムの場所）
- sysContact（システム連絡先）
- capability（機能の一意キー）
- customAttrName（カスタム属性名）
- customAttrValue（カスタム属性値）

フィルタには、AND、OR、およびグループ化（括弧）操作を含めることができます。詳細については、NNMi ヘルプの*「ノードグループ追加フィルタの指定」*を参照してください。

機能は、本来は NNMi と統合される他のプログラムを目的としていました。たとえば、ルーター冗長性とコンポーネント稼働状態は、機能（フィールド）を NNMi データベースに追加します。これらの機能は、すでに検出されてデバイスからノード詳細を調べることにより、見ることができます。

iSPI によりカスタム属性を追加したり、独自のカスタム属性を作成できます。Web Services SDK を購入していない方は、各ノードのフィールドに手動で値を入れる必要があります。たとえば資産番号やシリアル番号は属性となりえますが、機能ではありません。

追加ノード

フィルタを使用して制限するには難しすぎる重大デバイスがネットワークに含まれている場合、これらのデバイスをホスト名別でグループに追加できます。

インタフェースグループ

インタフェースグループは、ノード内のインタフェースを、IFType 別に、または ifAlias、ifDescr、ifName、ifIndex、IP アドレスなど他の属性別にフィルタリングします。インタフェースグループは階層もコンテインメントも継承しませんが、インタフェースをホスト管理しているノードのノードグループに基づいてメンバーシップをさらに限定することができます。

インタフェースグループを、ノードグループと同様のカスタム機能および属性でフィルタリングできます。

インタフェースグループの制限は、タブ内およびタブ間でまとめて AND を適用します。

ノード / インタフェース / アドレス階層

NNMi はモニタリング設定を、以下の方式で割り当てます。

1 **インタフェース設定** —NNMi は、ノードのインタフェースおよび IP アドレスの各々を、最初に一致する**インタフェース設定**定義に基づいてモニターします。最初に一致するのは、順序番号が最も小さい**インタフェース設定**定義です。

2 **ノードの設定** —NNMi は、各ノードと前回一致しなかった各々のインタフェースまたは IP アドレスを、最初に一致する**ノードの設定**定義に基づいてモニターします。最初に一致するのは、順序番号が最も小さい**ノードの設定**定義です。

子ノードグループは、順序階層に含まれます。親ノードグループの順序番号のほうが小さい場合（たとえば、親 =10、子 =20）、親ノードグループに指定されたモニタリング設定は子ノードグループ内のノードにも適用されます。親ノードグループモニタリング設定を上書きするには、子ノードグループの順序番号を親よりも小さな番号に設定します（たとえば、親 =20、子 =10）。

3 **デフォルト設定** — 手順 1 または 手順 2 でノード、インタフェース、または IP アドレスに一致が見つからない場合、NNMi はデフォルトのモニタリング設定を適用します。

すべてを停止して再度やり直す

検出を完全に再スタートして NNMi 設定のすべてのやり直したい場合、または NNMi データベースが破損した場合は、NNMi 設定およびデータベースをリセットできます。このプロセスにより、NNMi 設定、トポロジ、およびインシデントのすべてが削除されます。

この手順で説明しているコマンドの詳細は、該当する参照ページか UNIX のマンページを参照してください。

以下の手順に従ってください。

1 NNMi サービスを、次のコマンドを使用して停止します。

```
ovstop -c
```

2 （任意）この手順によってデータベースが削除されるため、実行する前に次のコマンドで既存のデータベースをバックアップするとよいでしょう。

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

3 Optional. 現在の NNMi 設定をとっておきたい場合は、`nnmconfigexport.ovpl` コマンドを使用して NNMi 設定を XML ファイルに出力します。

4 （任意）`nnmtrimincidents.ovpl` コマンドを使用して NNMi インシデントをアーカイブします。

5 NNMi データベースを削除して再作成します。

- 組み込みデータベースの場合は、次のコマンドを実行します。

  `nnmresetembdb.ovpl -nostart`

- Oracle データベースの場合は、Oracle データベース管理者に NNMi データベースの削除と再作成を依頼してください。データベースインスタンス名は、削除せずに保持してください。

6 iSPI または NNMi と統合されるスタンドアロン製品をインストールした場合は、これらの製品をリセットして古いトポロジ識別名を削除します。具体的な手順については、製品のマニュアルを参照してください。

7 NNMi サービスを、次のコマンドを使用して開始します。

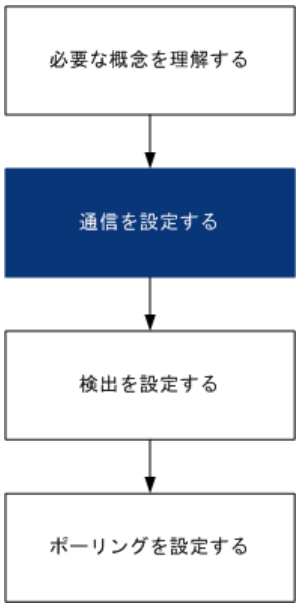
  `ovstart -c`

  これで NNMi はデフォルト設定のみとなり、本製品を新しいシステムにインストールしたのと同じ状態です。

8 NNMi の設定を開始します。次のいずれかを行います。

- 「クイックスタート設定ウィザード」を使用します。
- NNMi コンソールの **[設定]** ワークスペースに情報を入力します。
- `nnmconfigimport.ovpl` コマンドを使用して、手順 3 で保存した NNMi 設定の一部またはすべてをインポートしてください。

NNMi 通信



HP Network Node Manager i Software は、SNMP プロトコルと ICMP (ping) プロトコルの両方を使って、デバイスを検出し、デバイスのステータスと稼働状態をモニタリングします。各自の環境で実行可能な通信を確立するには、ネットワークのさまざまなデバイスとエリアについて、アクセス資格認定、適切なタイムアウト、再試行値すべてで NNMi を設定します。ネットワークのいくつかのエリアでプロトコルを無効にし、トラフィックを削減またはファイアウォールを順守できます。

設定する通信の値は NNMi の検出および状態ポーリングの基礎を形成します。NNMi は、設定された値を検出およびデータポーリング用のすべてのデバイスクエリーに適用します。このように、ネットワークのいくつかの領域への SNMP 通信を無効にすると、検出もポーリングも、SNMP トラフィックをその領域に送信できず、データを受信できません。

この章では、以下の内容を記載しています。

- 通信の概念
- 通信の計画作成
- 通信の設定
- 通信の評価
- 通信の調整

通信の概念

NNMi は、SNMP と ICMP をおもに要求と応答の方式で使います。管理ステーションは、簡単な応答性 (ICMP ping) を調べ、またはデバイス（SNMP MIB オブジェクト）が測定する特定のデータ値を要求して通信を開始します。

以下の概念が通信設定に適用されます。

- ネットワーク待ち時間とタイムアウト
- アクセス資格認定とコミュニティ文字列
- SNMP バージョンの優先
- 設定レベル

- 優先される管理アドレス
- ポーリングプロトコル

ネットワーク待ち時間とタイムアウト

どちらのプロトコルでも、通常のネットワーク待ち時間が、管理ステーションが応答を得るのにかかる時間に影響します。ネットワークの異なるエリアでは、応答が戻ってくるのにかかる時間も異なるのが普通です。管理ステーションがあるローカルネットワークでは、ほとんど即座に応答があります。ダイアルアップ広域リンク経由でアクセスする地理的に遠隔地域のデバイスからの応答は受信にはるかに長く時間がかかるのが普通です。

さらに、負荷が大きいデバイスは処理量が多いためクエリーにただちに応答できません。この待ち時間はネットワークの処理時間自体に追加する必要があります。

環境の各エリア、そして各デバイスまたはデバイスの種類に標準的なものに基づいて、どのくらいの時間 NNMi が応答を待機すべきか、何回試行すべきかを設定します。

NNMi は、再試行の設定を完了すると、繰り返しごとに以前のタイムアウト値に 2 を乗算します。

アクセス資格認定とコミュニティ文字列

SNMP 通信には、コミュニティ文字列（SNMPv3 ではユーザー）と呼ばれるアクセス制御のフォームが必要です。これの動作はパスワードにきわめて類似しています。管理ステーションからの各要求にはコミュニティ文字列が含まれており、管理対象デバイスはコミュニティ文字列が正しい場合にのみ応答します。

SNMPv1 と SNMPv2 では、コミュニティ文字列はクリアテキストとしてネットワークを通過するので、実際にはあまり安全ではありません。

SNMPv3 では、セキュリティが主要な目標になっています。設定された各ユーザーは、認証（パケットが記載されたユーザーから本当に MD5 または SHA 経由で送信されたことを証明）とオプションでプライバシー（DES による暗号化）を許可するプロファイルを持っています。マネージャからの通信には正しいユーザーが含まれている必要があるのみでなく、そのユーザーについて設定された認証およびプライバシー制御に従う必要もあります。

NNMi では、ノードグループに使われている可能性のある複数のコミュニティ文字列またはユーザーを設定できます。設定したすべての値を並行して試し、グループ内のデバイスと通信しようとします（指定のセキュリティレベルでです。下記参照）。デバイスから応答を返す最初の値が、検出とモニタリングに使われる資格認定になります。デバイスを再設定したために資格認定が作用を停止した場合、NNMi は、設定された複数の値を再試行し、今の時点で作用する 1 つの値を見つけます。

デフォルトでは、設定された資格認定についてデバイスからの SNMP 応答がない場合、デバイスは検出されず、データベースに追加されません。しかし、このデフォルトは上書きできます。その場合、デバイスは `No SNMP` デバイスプロファイルで検出される可能性があります。

SNMP バージョンの優先

SNMP プロトコルはバージョン 1 からバージョン 2(c) へと長年をかけて発展したもので、現在はバージョン 3 です。この間、とりわけセキュリティ機能は強化されてきました。NNMi は、各自の環境でどのバージョンでも処理できますし、全バージョンの混合したものも処理できます。

設定の一部として、ネットワークの各エリアで受け入れられるセキュリティの最低レベルを NNMi に通知します。セキュリティを最低レベルに制約する場合、NNMi は、以下を実行して、プロトコルの優先されるバージョンを選択します。どの時点でも最初の応答が、通信資格認定と使用されるバージョンを設定し、評価は停止します。

1. 1 つまたは複数のコミュニティ文字列が設定される場合は、タイムアウトと再試行に設定された値で SNMPv2 を使って通信しようとしてください。
2. SNMPv2 を使ったコミュニティ文字列への応答がない場合は、タイムアウトと再試行について設定された値で SNMPv1 を使って通信をしようとしてください。
3. 1 つまたは複数の SNMPv3 ユーザーが設定されている場合は次のとおりです。
   a. No Authentication（認証なし）、No Privacy（プライバシーなし）のユーザーは、タイムアウトと再試行に設定された値で SNMPv3 を使って通信しようとしてください。
   b. Authentication（認証あり）、No Privacy（プライバシーなし）のユーザーは、タイムアウトと再試行に設定された値で SNMPv3 を使って通信しようとしてください。
   c. Authentication（認証あり）、Privacy（プライバシーあり）のユーザーは、タイムアウトと再試行に設定された値で SNMPv3 を使って通信しようとしてください。
4. デフォルトでは、設定された資格認定についてデバイスからの SNMP 応答がない場合、デバイスは検出されず、データベースに追加されません。しかし、これは上書きできます。デバイスは `No SNMP` デバイス プロファイルで検出される可能性があります。

設定レベル

NNMi では、以下のレベルで、アクセス資格認定、タイムアウトと再試行、プロトコルの使用可能性すべてを設定できます。

- 特定のノード
- 領域
- グローバルデフォルト

各レベルで、アクセス資格認定、タイムアウトと再試行の値、プロトコル使用可能性すべてを設定できます。あるレベルで設定をブランクにし、次のレベルでデフォルトを適用することもできます。

NNMi は、指定 IP アドレスで通信を使うとき、次のように構成設定を適用します。NNMi

1. アドレスが特定のノードの設定と一致する場合は、利用できる値をすべて記入します。設定がブランクのところはブランクのままにします。
2. まだブランクがある場合は、アドレスを評価してそれが領域定義と一致するかどうか調べます。領域は重なる可能性があるので、順序番号が最小のものと一致するものを使います。領域レベルで使用可能な値のあるブランクには記入します。
3. それでもブランクがある場合は、グローバルデフォルト設定からそれらに記入します。（追加領域と一致するものは使用しません）。

このように、全ブランクに記入が行われるまで、より高いレベルのデフォルトを調べ続け、適用する値を累積的に構築します。

優先される管理アドレス

ルーターのようなデバイスには、多数の IP アドレスが割り当てられます。自動検出ルールを使う場合、NNMi は他のデバイスの ARP キャッシュ内の IP アドレスの 1 つが存在するルーターを検出します。次に、NNMi はすべてのルーターのインタフェースを検出します。そして、デバイスとのそれ以上の全通信について**管理 IP アドレス**として使う 1 つのアドレスを選択します。NNMi が管理 IP アドレスを決定する方法については、NNMi ヘルプの「*ノードのフォーム*」を参照してください。

NNMi の選択済み管理アドレスが、優先するアドレスと一致しない場合は、上書きして NNMi が自分の設定するアドレスを使うようにできます。

ベストプラクティス

NNMi のルールの 1 つは、最初に検出された IP アドレスを管理アドレスとして使用することを指定しています。優先される IP をシードアドレスとして設定して、NNMi の選択に影響を与えることができます。シード選択の指示については、検出シードページの 51 を参照してください。

ポーリングプロトコル

ファイアウォールがアクセスをブロックする場合は、SNMP または ICMP 用にネットワークの一部へのアクセスを無効にできます。

ネットワークのある領域への ICMP トラフィックを無効にすると、次のような結果になります。

- 自動検出ルール ping スイープ機能は、追加ノードを見つけられません。すべてのノードが、シードされるか、または ARP キャッシュまたは検出プロトコル（CDP や EDP）MIB エントリによって利用できる必要があります。広域ネットワーキングデバイスは、すべてシードしないと失われる可能性があります。
- 状態ポーリングは、SNMP アクセスのないデバイスをモニタリングできません。ICMP 障害ポーリンググループを設定することはできますが、デフォルトでは、State Poller が ICMP を使うのはこのときのみです。
- オペレータはトラブルシューティングの間は、**[アクション] > [Ping]** を使ってデバイス到達可能性をチェックできません。

ネットワークのある領域への SNMP トラフィックを無効にすると、次のような結果になります。

- 検出では、存在しないデバイスの情報は収集できません。すべてのデバイスに `No SNMP` デバイスプロファイルがあります。
- 検出では、クエリーによって追加デバイスを知ることができません。デバイスはすべて直接にシードされる必要があります。
- 検出では、データベースから接続情報を収集できないので、デバイスは未接続として示されます。
- State Poller は、`No SNMP` デバイス用の設定を使います。また、デフォルトで、ICMP (ping) を使ってデバイスの可用性をモニタリングします。
- State Poller は、デバイスからコンポーネント稼働状態またはパフォーマンスを収集することはできません。
- Causal Engine は、デバイスに接触して近隣分析を実行し、インシデントの根本分析を見つけることはできません。

通信の計画作成

次のエリアで意思決定を行います。

- 意味のあるデフォルト
- 通信設定領域
- 特定のノードの設定
- 再試行とタイムアウトの値
- アクティブなプロトコル
- 複数のコミュニティ文字列またはアクセス資格認定

意味のあるデフォルト

評価の最後で、デフォルト値を使ってブランクに記入するので、デフォルト値はネットワークの大部分に合理的なものに設定してください。

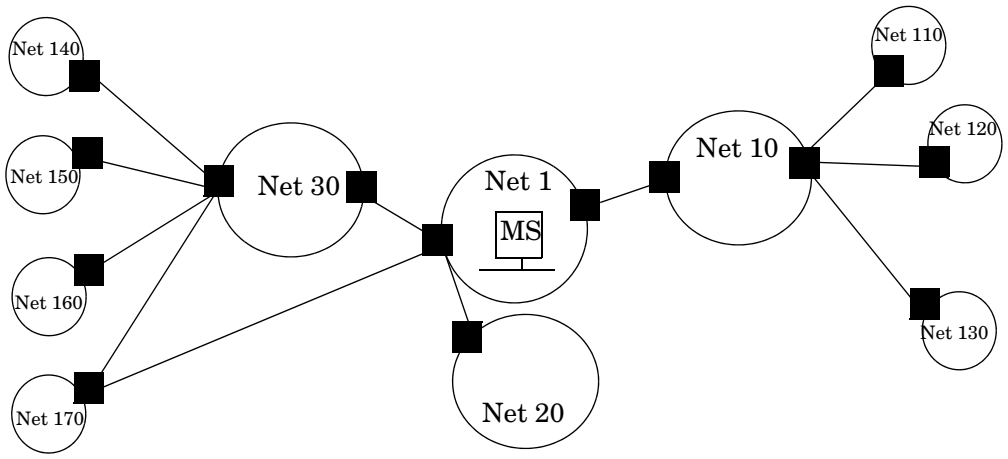
- NNMi が試す必要のある一般に使われるコミュニティ文字列がありますか？
- 他の設定が該当しない場合、ネットワークではどんなタイムアウト値と再試行値が合理的でしょうか？

領域または特定のノードのレベルでは、他の値が設定されます。

通信設定領域

領域とは、類似の通信設定が意味をなすネットワークエリアのことです。たとえば、管理ステーションの周りのローカルネットワークはきわめて素早く応答を返します。複数ホップ離れたネットワークエリアなら応答にもっと時間がかかるのが普通です。

ネットワークのすべてのサブネットまたはエリアを設定する必要はありません。類似のタイムラグでこれらを結合できます。次のネットワークマップについて考えてみてください。



タイムアウト / 再試行の目的では、Net 1 について領域を設定することもできます。これは、Net 10、Net 20、Net 30 を含む領域であり、もっと離れた遠いネットワーク用の領域です。管理サーバーから 1 ホップまたは 2 ホップのパスを優先するようトラフィック管理構成が設定されているかどうかに従って、Net 170 をグループにまとめる最良の方法を決定します。

また、領域を使って、類似のアクセス資格認定を使うデバイスがグループにまとめられます。ネットワークの全ルーターが同じコミュニティ文字列（または可能な文字列の小さいセット）を使っており、しかも `rtr`*nnn*`.yourdomain.com` のような命名規約でルーターを識別できる場合は、ルーターが同じように処理されるように領域を設定できます。ワイルドカードを使ってデバイスをグループにまとめられない場合は、デバイスを特定のノードとして設定できます。

これで、同じタイムアウト / 再試行の値とアクセス資格認定設定を領域の全ノードに適用できるように、領域設定の計画を作成できます。

領域定義は重なることがありますし、デバイスは複数の領域にあてはまることがあります。NNMi は、順序番号の最も小さい領域の設定を適用します。

特定のノードの設定

特定のパラメータのため、またはワイルドカード機能がないため領域定義と一致しないデバイスについては、ノードについて使われる特定のノードの設定と値が必要であることをドキュメントに記載してください。

再試行とタイムアウトの値

タイムアウトの時間を長く、再試行の回数を多く設定すると、ビジー状態にあるか、または離れたところにあるデバイスからより多くの応答を集められます。これによって偽のダウンメッセージを除外できます。しかし、実際にダウンしているデバイスに注意が必要なことを知るのに時間がかかるようにもなります。ネットワークの各エリアのバランスを見出すことは重要であり、このために各自の環境で値のテストと調整の期間が必要になる可能性があります。

各ホップに関する現在のラグタイムの発想を得るには、各ネットワークエリアのデバイスに `traceroute` を実行します。

アクティブなプロトコル

デバイスに向かうトラフィックの種類を制御するために、通信レイヤーとモニタリング設定の 2 つの機会があります。インフラストラクチャのファイアウォールによってトラフィックの種類が差し止められるときは、通信レイヤーを使います。デバイスのセットにその種類のデータが不要なときは、モニタリング設定を使います。どちらのレイヤーでもプロトコルが無効な場合、そのトラフィックはデバイスに送信されません。

各領域または特定のデバイスは ICMP トラフィックを受信するはずであるかに注意してください。

SNMP 通信を無効にすると、ネットワークのステータスと稼働状態をモニタリングする NNMi の機能がかなり危険にさらされます。

アクセス資格認定を与えないデバイスとの SNMP 通信を明示的に無効にする必要はありません。デフォルトで、NNMi はこれらのデバイスを `No SNMP` デバイスプロファイルに割り当て、ICMP のみを使ってデバイスをモニタリングします。

複数のコミュニティ文字列またはアクセス資格認定

グローバルなデフォルトおよび各領域について、試みる複数のコミュニティ文字列を並行して設定できます。

有望なコミュニティ文字列を試す間に、NNMi がデバイスに資格認定不合格を生成させることがあります。NNMi が初期検出を完了する間に、資格認定不合格は安全に無視できる可能性があることを業務部に知らせてください。

代わりに、領域とコミュニティ文字列が可能な限り厳しく試行するように設定して、資格認定不合格の数を最小にすることもできます。

環境で SNMPv1/2 と SNMPv3 の両方が使用されている場合は、各領域で受け入れられる最低のセキュリティレベルならびに各レベルで試みる資格認定を決定してください。

ネットワークの各エリアで試みるアクセス資格認定の計画を作成します。

通信の設定

この項を読んだ後、特定の手順については、NNMi ヘルプの「*通信プロトコルを設定する*」を参照してください。

大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。詳細については、ベストプラクティス : 既存の設定を保存ページの 28 を参照してください。

通信の以下のエリアの設定

- デフォルト設定
- 領域定義とその設定
- 特定のノードの設定

  特定のノードについて、NNMi コンソールまたは構成ファイルによって、ノードの設定を入力できます。

すべての［**通信の設定**］を［**保存して閉じる**］と、NNMi コンソールに戻り、変更が導入されます。

ベストプラクティス　複数のグループにあてはまるオブジェクトは順序番号の最も小さいグループから設定を適用されることを頭に入れておきながら、順序番号をダブルチェックします。

通信の評価

この項では、通信設定の進行と成功を評価する方法をリストします。これらの作業のほとんどを完了できるのは、検出が完了した後です。

以下について考えます。

- すべてのノードが SNMP 用に設定されましたか？
- デバイスについて SNMP アクセスは現在利用できますか？
- NNMi は正しい通信設定を使っていますか？

すべてのノードが SNMP 用に設定されましたか？

1 ［**ノード**］インベントリビューを開きます。

2 No SNMP という文字列を含む **Device Profile** コラムをフィルタします。

- 管理するデバイスごとに、特定ノードの通信設定を行います。その代わりに、領域を拡張して、ノードを組み入れ、アクセス資格認定を更新することもできます。
- 通信設定が正しい場合は、デバイスの SNMP エージェントが実行中であり、適切に設定されていることを確認します（ACL を含みます）。

デバイスについて SNMP アクセスは現在利用できますか？

1 インベントリビューでノードを選択します。

2 ［**アクション**］>［**ステータスポーリング**］または［**アクション**］>［**ポーリングの設定**］を選択します。

結果に SNMP の値が表示された場合、通信は動作中です。

コマンドラインから nnmsnmpwalk コマンドで通信をテストすることもできます。詳細は、*nnmsnmpwalk* リファレンスページまたは UNIX のマンページを参照してください。

NNMi は正しい通信設定を使っていますか？

SNMP コミュニティ文字列が欠落しているか、または正しくない場合は、検出が不完全になる可能性がありますし、検出パフォーマンスに悪影響を及ぼす可能性もあります。

デバイスの設定を確認するには、nnmcommconf.ovpl コマンドを使用するか、または以下の手順を実行します。

1 インベントリビューでノードを選択します。

2 ［**アクション**］>［**通信設定**］を選択します。

NNMi は、表示された値を求めるために、特定のノード一致、順序番号による領域設定、デフォルト設定をすべて評価します。

nnmcommconf.ovpl コマンドの出力例は次のとおりです。

```
node1.domain.com についての SNMP 設定
name = node1.domain.com
address = 10.2.100.6
addressForced = false
getCommunity = [public]
timeout = 5000
retries = 1
port = 161
enabled = true
region name = Core Network
```

この出力は、現在のタイムアウト値と再試行値、通信用の IP アドレス、および管理者アクセスがある場合は設定されたアクセス資格認定を示します。

getCommunity と region name に関する NNMi の値が正しいことを確認します。getCommunity 文字列は、SNMP エージェントが使う読み取り専用コミュニティ文字列と一致する必要があります。region name は、このノードが一致すると予想する自動検出ルールの名前と一致する必要があります。

ノードの最も新しい検出から実際に使われている値を知る手順は次のとおりです。

1 **[ノード]** フォームを開きます。

2 **[SNMP エージェント]** フィールドで、**[SNMP エージェント]** フォームを開きます。

フォームには、プロトコルのバージョンおよびその他の SNMP 設定が表示されます。

State Poller 設定は通信設定と一致していますか？

通信設定によってネットワークのエリアへのプロトコルトラフィックが許容される場合でも、その種類のトラフィックはモニタリング設定で無効にされることがあります。設定が上書きされるかどうかを知る手順は次のとおりです。

1 インベントリビューでノードを選択します。

2 **[アクション]** > **[モニタリング設定]** を選択します。

モニタリング設定または通信設定のどちらかによってデバイスへのある種類のトラフィックが無効にされる場合、そのトラフィックは NNMi から送信されません。

管理 IP アドレスは正しいですか？

1 インベントリビューでノードを選択します。

2 **[アクション]** > **[通信設定]** を選択します。

通信の調整

認証不合格の削減

検出の間に NNMi があまりにも多くの認証トラップを生成している場合は、NNMi が試行するアクセス資格認定の、より小さいグループで小さい領域または特定のノードを設定します。

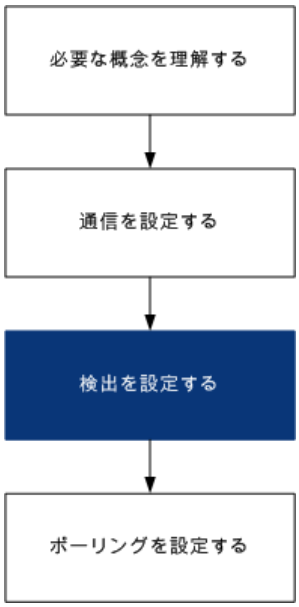
タイムアウトと再試行の調整

NNMi が検出の間に SNMP を使ってデバイスに接触しようとするとき、通信設定は NNMi が必要なデバイス情報を収集できるか調べます。通信設定に正しい SNMP コミュニティ文字列が含まれていない場合、または NNMi が非 SNMP デバイスを検出している場合、NNMi は SNMP タイムアウトと再試行用に設定済みの構成を使います。この場合、タイムアウトの値が大きいか、または再試行の回数が多いと、検出の全般的パフォーマンスに悪影響が及ぶ可能性があります。SNMP/ICMP 要求に低速で応答することが分かっているデバイスがネットワークにある場合は、**[通信設定]** フォームの **[領域]** タブまたは **[特定のノードの設定]** タブを使ってこれらのデバイスについてのみタイムアウト値と再試行値を微調整することを考えてください。

デフォルトコミュニティ文字列の削減

デフォルトコミュニティ文字列が多数あると、検出パフォーマンスに悪影響が及ぶことがあります。多数のデフォルトコミュニティ文字列を入力する代わりに、**[通信設定]** フォームの **[領域]** タブまたは **[特定のノードの設定]** タブを使って、ネットワークの特定エリアのコミュニティ文字列設定を微調整します。

NNMi 検出



ネットワーク管理で最も重要な作業の一つは、常に最新のネットワークトポロジを把握しておくことです。HP Network Node Manager i Software 検出により、トポロジインベントリにネットワーク内のノードに関する情報が投入されます。NNMi は、継続的なスパイラル検出によってこのトポロジ情報を維持します。これにより、根本原因解析ツールとトラブルシューティングツールで、インシデントに関する正確な情報を把握できるようになります。

この章では、NNMi 検出を設定するために役立つ情報を記載しています。検出がどのようにして行われるのかと検出の設定方法については、NNMi ヘルプの*「ネットワークの検出」*を参照してください。

NNM 6.x/7.x の使用経験があり NNMi 8.10 で検出がどのように変わったのかを知りたい方は、ネットワーク検出ページの 136 でこの両者の違いについての高度な説明をお読みください。

この章では、以下の内容を記載しています。

- 検出の概念
- 検出の計画
- 検出の設定
- 検出の評価
- 検出の調整

検出の概念

NNMi のルーターとスイッチのみを検出するデフォルト動作では、ネットワーク管理を最も重要なデバイスに集中させることができます。つまり、最初にネットワークの基幹をターゲットにします。一般に、末端ノード（たとえばパソコンやプリンタ）を管理対象にするのは、それらを重大リソースと見なすのでない限り避けるべきでしょう。たとえば、データベースやアプリケーションサーバーがクリティカルなリソースとして考えられます。

NNMi で検出するデバイスを管理して NNMi トポロジに加えるには、いくつかの方法があります。ネットワークをどのように構成するかや NNMi で何を管理するかによって、検出構成を非常に単純にしたり、極めて複雑にしたり、その間の適当なレベルにできます。

NNMi では、事前設定をせずに行える検出はありません。各種のデバイスが NNMi トポロジに存在する前に、検出を設定する必要があります。

検出された各ノードは、NNMi が積極的にそのノードを管理しているかどうかに無関係に、ライセンスの最大数まで数えられます。所有している NNMi ライセンスの内容は、検出方法にも影響を及ぼします。

ステータスモニタリングの考慮事項も、選択肢に影響を及ぼします。State Poller は、デフォルトでは NNMi が検出したデバイスに接続したインタフェースしかモニタリングしません。ネットワークのいくつかの領域ではこのデフォルト設定を変更できるため、職責の範囲を超えたデバイスの検出が可能になります。（State Poller の詳細は、NNMi 状態ポーリングページの 57 を参照してください。）

NNMi には、次の 2 つの基本的な検出設定モデルがあります。

- **リストベース検出** —NNMi に、どのデバイスをデータベースに追加してモニタリングするかを、シードのリストによって明確に指示します。
- **ルールベース検出** —NNMi にネットワークのどの領域とデバイスタイプをデータベースに追加するかを伝え、NNMi に各領域の開始アドレスを指定して、NNMi に定義されたデバイスを検出させます。

リストベース検出とルールベース検出を自由に組み合わせて、NNMi の検出対象を設定できます。初回の検出によってこれらのデバイスが NNMi トポロジに追加され、スパイラル検出ではネットワークが日常的に再検出されるため、トポロジは常に最新の状態が維持されます。

NNMi はデバイスプロファイルから属性を導き出す

NNMi はデバイスを検出する際に、SNMP を使用していくつかの属性を直接収集します。重要な属性の 1 つは MIB II システムオブジェクト ID (`sysObjectID`) です。システムオブジェクト ID から、NNMi はベンダー、デバイスカテゴリ、デバイスファミリーなどの追加属性を導き出します。

検出中、NNMi は MIB II システムの性能を収集して、データベースのトポロジ部分に格納します。システム性能は、**「ノード詳細」**フォームで見ることができます。ただし、これらの性能は NNMi の他の部分（つまり、モニタリング設定）では使用されません。NNMi では、デバイスカテゴリ（システムオブジェクト ID のデバイスプロファイルにより）を使用して、デバイスをノードグループに分類します。ノードビュー表では、**「デバイスカテゴリ」**列に各ノードのデバイスカテゴリが明示されます。

NNMi では、リリース時に公開された 3600 以上のシステムオブジェクト ID のデバイスプロファイルが付属しています。ご使用の環境内にしかないデバイス用にデバイスプロファイルをカスタム設定して、これらのデバイスをカテゴリ、ベンダーなどに対応付けることができます。

検出の計画

以下の領域で決定を下してください。

- 基本的な検出方法を選択する
- 自動検出ルール
- ノード名の解決
- サブネット接続ルール
- 検出シード
- 再検出の間隔
- オブジェクトを検出しない

基本的な検出方法を選択する

完全なリストベース検出を行うのか、完全なルールベース検出を行うのか、それともこの 2 つの方法を組み合わせて使用するのかを決定します。

リストベース検出

リストベース検出では、NNMi で検出する各ノードを（検出シードとして）明確に指定します。

リストベース検出のみを使用することの利点を以下に示します。

- NNMi の管理対象を厳密に管理できます。
- 設定が最も簡単です。
- 固定的なネットワークに適しています。
- NNMi を初めて使用する場合に適した方法です。自動検出ルールを、徐々に追加していくことができます。

リストベース検出のみを使用することのデメリットを以下に示します。

- NNMi は、ネットワークに新規ノードが追加されても検出しません。
- 検出対象とするノードのリストを指定しなければなりません。

ルールベース検出

ルールベース検出では、NNMi が検出して NNMi トポロジに入れるネットワークの領域を定義するために 1 つ以上の自動検出ルールを作成します。各々のルールに対して、1 つ以上の検出シードを（シードを明確に指定するか ping スィープを有効にすることにより）指定する必要があります。それにより NNMi がネットワークを自動的に検出します。

ルールベース検出を使用することの利点を以下に示します。

- 大規模なネットワークに適しています。NNMi は大量の数のデバイスを、最低限の設定項目に基づいて検出できます。
- 頻繁に変わるネットワークに適しています。ネットワークに追加した新しいデバイスは、管理者が介在しなくても検出されます（各デバイスは自動検出ルールの適用範囲内であることが前提）。

- 新規デバイスをタイミングよく管理するためのサービス内容合意書や、許可されていない新規デバイスがあれば注意を与えるためのセキュリティガイドラインを順守するために、新しいデバイスがネットワークに追加されると検出されます。

ルールベース検出を使用することのデメリットを以下に示します。

- すぐにライセンス限度に達してしまいます。
- ネットワークの構造によっては、自動検出ルールの調整が複雑になることがあります。
- 自動検出ルールが非常に広範囲で管理したい数以上のデバイスを NNMi が検出する場合、不要なデバイスを NNMi トポロジから削除しなければならなくなります。NNMi には一括削除のためのツールはありません。そのため、広範囲な検出から除去するのは非常に困難です。

ルールベース検出のみ

自動検出ルール

自動検出ルールの順序

自動検出ルールの**順序**属性の値は、検出範囲に次のように影響します。

- IP アドレス範囲

  デバイスが 2 つの自動検出ルールに該当すると、順序番号が小さい方の自動検出ルールの設定が適用されます。たとえばある自動検出ルールにより IP アドレスの一式が除外されると、それより大きな順序番号の自動検出ルールはこれらのノードを処理せず、そのアドレス範囲内のノードは、検出シードとしてリストされない限り検出されません。

- システムオブジェクト ID の範囲
  - 自動検出ルールに IP アドレス範囲が含まれていない場合は、システムオブジェクト ID の設定が、それより大きな順序番号のすべての自動検出ルールに適用されます。
  - 自動検出ルールに IP アドレス範囲が含まれている場合、システムオブジェクト ID 範囲は自動検出ルール内でのみ適用されます。

デバイスを検出から除外

- 特定のオブジェクトタイプが検出されないようにするには、検出したくないシステムオブジェクト ID を無視する自動検出ルールを、順序番号を小さくして作成します。このルールに IP アドレス範囲を含めないでください。この自動検出ルールに小さい順序番号を付けることで、このルールに一致するオブジェクトを検出プロセスはすぐにとばします。
- IP アドレス範囲またはシステムオブジェクト ID 範囲の**ルールにより無視**設定は、その自動検出ルールのみに影響します。無視される範囲内に含まれるデバイスは、別の自動検出ルールに含めることが可能です。
- NNMi 8.10 の新機能。**[検出の設定]** フォームの **[除外対象の IP アドレス]** タブでリストされるアドレスは、すべての自動検出ルールに適用されます。これらのアドレスは検出シードとして設定されない限り、NNMi トポロジには追加されません。(検出シードは常に検出されます。)

Ping スィープ

NNMi 8.10 の時点で、ping スィープを使用して、設定された自動検出ルールの IP アドレス範囲内のデバイスを検出できます。初期検出では、すべてのルールで ping スィープを有効にするとよいでしょう。そうすることで十分な情報が NNMi 検出に提供されるので、検出シードを設定する必要がなくなります。

➤ ping スィープは、16 ビット以下のサブネット（たとえば `10.10.*.*`）で機能します。

ping スィープは特に、ISP ネットワークのように制御が不要な WAN 全体でのデバイスの検出で便利です。

➤ ファイアウォールは ping スィープをネットワークに対する攻撃としてみなすことがよくあり、その場合、ファイアウォールは ping スィープを発信したデバイスからのすべてのトラフィックをブロックすることがあります。

ベストプラクティス　ping スィープは、小さな検出範囲にのみ有効にしてください。

自動検出ルールの検出シード

自動検出ルールごとに少なくとも 1 つの検出シードを指定してください。検出シードを指定するためのオプションを以下に示します。

- **［検出の設定］**フォームの**［検出シード］**タブでシードを入力します。
- `nnmloadseeds.ovpl` コマンドを使用して、シードファイルから情報をロードします。
- 少なくとも初回の検出で、ping スィープをルールに対して有効にします。

自動検出ルールのベストプラクティス

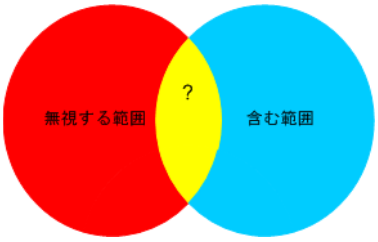
- NNMi はすべての検出対象デバイスを自動的に管理するため、管理したいネットワークの範囲に厳密に一致する IP アドレス範囲を使用してください。
  - 複数の IP アドレス範囲を 1 つの自動検出ルール内で使用して、検出を限定することができます。
  - 自動検出ルールに大きな IP アドレス範囲を追加した後に、そのルール内の検出からいくつかの IP アドレスを除外することができます。
- システムオブジェクト ID 範囲の指定は接頭部分であり、絶対値ではありません。たとえば、範囲 1.3.6.1.4.1.11 は 1.3.6.1.4.1.11.* と同じです。

例

検出ルールの重複

図 1 は、重複する 2 つの検出範囲を示しています。左側の円は、NNMi 検出で無視される IP アドレス範囲またはシステムオブジェクト ID 範囲を表しています。右側の円は、NNMi 検出で検出されて含まれる IP アドレス範囲またはシステムオブジェクト ID 範囲を表しています。重複している領域は、これらの自動検出ルールの順序に応じて検出に含まれるか無視されます。

図 1　重複している検出範囲



デバイスタイプ検出を制限する

ネットワーク内のプリンタ以外のすべての HP デバイスを検出するには、HP エンタープライズシステムオブジェクト ID (1.3.6.1.4.1.11) を含む範囲を持つ 1 つの自動検出ルールを作成します。この自動検出ルールで、HP プリンタ (1.3.6.1.4.1.11.2.3 9) のシステムオブジェクト ID を無視する 2 番目の範囲を作成します。IP アドレス範囲を未設定のままにしてください。

ノード名の解決

デフォルトでは、NNMi はノードを次の順序で識別しようとします。

1. 短い DNS 名
2. 短い sysName
3. IP アドレス

以下のシナリオでは、ノード名解決のデフォルト順序を変更したほうがよい場合を説明しています。

- 組織が DNS 設定の更新を外部者にまかせている場合、ネットワークに新しいデバイスが追加されるごとにその sysName を定義するポリシーを設定できます。この場合、sysName の選択をノード名解決の最初の選択肢として設定して、新しいデバイスがネットワークに導入されるとすぐに NNMi が検出できるようにします。（sysName を、そのデバイスを使用している間は維持します。）
- 組織が管理対象デバイスの sysName を設定も維持もしない場合、sysName をノード名解決の 3 番目の選択肢として選択します。

ベストプラクティス　DNS 完全名または DNS 短縮名を基本的な命名方法として使用している場合、NNMi 管理サーバーからすべての管理対象デバイスへの順方向と逆方向の DNS 解決があることを確認してください。

DNS 完全名が命名方法の場合、トポロジマップ上のラベルを長くできます。

ベストプラクティス　NNMi 8.10 の時点で、NNMi は最小のループバックアドレスを Cisco デバイスの管理アドレスとして選択するため、各 Cisco デバイスの最小のループバックアドレス上に DNS 解決を配置してください。一方、NNMi 8.0x は最大のループバックアドレスを管理アドレスとして選択します。

サブネット接続ルール

リストベース検出のみ

リストベース検出では、NNMi はサブネット接続ルールを使用して WAN 上の接続を検出します。NNMi は予測される接続の各末端で検出したデバイスのサブネットメンバーシップを評価し（IP アドレスとサブネット接頭部を調べて）、サブネット接続ルールで一致があるか調べます。

ルールベース検出のみ

自動検出ルールが有効で NNMi が /28 と /31 の間のサブネット接頭部で設定されたデバイスを見つけると：

1. NNMi は適用可能なサブネット接続ルールについて調べます。
2. 一致が見つかると、NNMi はサブネット内の有効な各アドレスをヒントとして使用して、そのアドレスでの検出を試みます。

ベストプラクティス

デフォルトの接続ルールを使用してください。問題がある場合のみそれらを変更してください。

検出シード

検出シードとして使用するデバイスをリストします。

ベストプラクティス

優先管理 IP アドレスを選択するための NNMi のルールの 1 つは、最初に検出した IP アドレスを管理アドレスとして使用することを指定しています。優先 IP アドレスをシードアドレスとして設定することにより、NNMi に影響を与えることができます。

ベストプラクティス

Cisco デバイスの場合、ループバックアドレスを検出シードとして使用してください。ループバックアドレスが、デバイス上の他のアドレスより確実に到達可能であるためです。DNS が、デバイスホスト名からループバックアドレスを解明するように正しく設定されていることを確認します。

リストベース検出のみ

リストベース検出の場合、NNMi の管理対象にするすべてのデバイスをリストします。このリストを、資産管理ソフトウェアから、または他のツールからエクスポートすることが可能です。

NNMi はこのリストにデバイスを自動的に追加することがないため、責任を負っているデバイスだけがリストに追加含まれるようにするか、モニタリング / ステータス計算に影響を及ぼすデバイスだけがリストに含まれるようにしてください。

ルールベース検出のみ

NNMi 8.10 の時点で、検出シードはルールベース検出のオプションです。

- ping スィープが自動検出ルールに対して有効の場合、そのルールのシードを指定する必要はありません。
- ping スィープが無効な各自動検出ルールで、ルールごとに少なくとも 1 つのシードを確認してください。ルールに IP アドレス範囲が複数含まれる場合、ルーターは WAN リンク全体の ARP エントリを維持しないため、それぞれのルーティング可能範囲でシードが必要になります。

ベストプラクティス

ルールベース検出を最も完璧なものにするためには、スイッチではなくルーターを検出シードとして使用してください。一般にルーターはスイッチより大きな ARP キャッシュを持っているためです。検出したいネットワークにコアルーターが接続されていれば、検出シードとしては最適な選択肢になります。

再検出の間隔

NNMi は、データベース内の各デバイスの設定情報を、設定された再検出間隔に従って再チェックします。さらに、NNMi は自動検出ルールの対象となる各ルーターから ARP キャッシュを収集して、ネットワーク上に新しいノードがあるか調べます。

デバイスの通信関連の設定に、インタフェースの番号変更のような変更があると、NNMi は自動的に、そのデバイスとその隣接デバイスに関するデータを更新します。

次のような変更では自動再検出は行われません。デバイスは設定された再検出間隔に基づいて更新されます。

- ノード内の変更（たとえば、ファームウェアアップグレードまたは接点システム）。
- ネットワークに追加された新しいノード。

ネットワーク内の変更のレベルに合った再検出間隔を選択します。非常に動的なネットワークでは、最低 24 時間の間隔を使用するとよいでしょう。これより安定したネットワークでは、その期間を広げることができます。

オブジェクトを検出しない

NNMi では、NNMi が特定のオブジェクトを無視するように設定する 3 つの方法があります。

- **［通信の設定］**フォームで、ICMP または SNMP 通信（あるいはその両方）を、グローバルで、通信領域で、または特定のホスト名または IP アドレスなどさまざまなレベルでオフにできます。これらのプロトコルの一方、あるいは両方を無効にすることの影響の詳細は、ポーリングプロトコルページの 38 を参照してください。
- **［検出の設定］**フォームで、NNMi に特定の IP アドレスや SNMP システムオブジェクト ID からヒントを収集しないように指示する自動検出ルールを設定できます。この基準に一致するノードはマップとデータベース上で存在し続けますが、スパイラル検出はこれらの IP アドレスまたはオブジェクトタイプを超える隣接デバイスまで行われません。
- **［検出の設定］**フォームで、NNMi に特定の IP アドレス範囲または特定の IP アドレス（あるいはその両方）をデータベースから除外するよう指示する自動検出ルールを設定できます。スパイラル検出では、あらゆるノードのアドレスリストでこれらのアドレスを表示したり、デバイス間に接続を確立するときこれらのアドレスを使用することがないので、NNMi がこれらのアドレスの使用状況をモニタリングすることはありません。

検出の設定

ここでは、設定のヒントを一覧にし、いくつかの設定例について説明します。このセクションの情報を読んだ後で、NNMi ヘルプの*「検出の設定」*で具体的な手順を参照してください。

NNMi は、**［検出シード］**フォームを**保存して閉じる**とすぐにシードから検出を開始するので、シードを設定する前に次のことを必ず行ってください。

- すべての通信設定を完了する。
- すべての自動検出ルール（ある場合）を完了する。
- サブネット接続ルールを設定する。
- 名前解決設定を設定する。
- **保存して閉じる**をコンソールまでさかのぼって行う。

大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。詳細については、ベストプラクティス：既存の設定を保存ページの 28 を参照してください。

自動検出ルールを設定する場合のヒント

- 新しい自動検出ルールを定義するときは、それぞれの設定を慎重に確認してください。新しいルールでは、自動検出はデフォルトで有効になっており、IP アドレス範囲はデフォルトで含まれており、システムオブジェクト ID 範囲はデフォルトで*無視*されます。

シードを設定する場合のヒント

- 検出対象ノードがリストされたファイルがすでにある場合は、この情報をシードファイルとして書式設定して、`nnmloadseeds.ovpl` コマンドを使用してそのノードリストを NNMi にインポートします。
- シードファイルで、管理アドレスとして NNMi が選択する IP アドレスに影響を与える手段として IP アドレスを指定します。(ホスト名を使用すると、DNS は IP アドレスを各ノードに提供します。)
- シードファイルの各エントリに適切な書式を、以下に示します。

  ```
  IP_address # node name
  ```

  この書式は、NNMi と人間の両方が容易に理解できます。
- 保守目的のため、使用するシードファイルは 1 つだけにすることをお勧めします。ノードを必要に応じて追加して、`nnmloadseeds.ovpl` コマンドを再度実行します。NNMi は新しいノードを検出しますが、既存のノードは再判定しません。
- ノードをシードファイルから削除しても、NNMi トポロジからは削除されません。ノードは直接 NNMi コンソールで削除してください。
- ノードをマップやインベントリビューから削除しても、シードは削除されません。
- NNMi でノードを再検出したい場合は、そのノードをマップまたはインベントリビューから、*そして*NNMi コンソールの **[検出の設定]** フォームの **[検出シード]** タブから削除してから、そのノードを NNMi コンソールで入力するか、`nnmloadseeds.ovpl` コマンドを実行します。

ルールベース検出のみ

- 検出ルールを、そのルールのシードを指定する*前に*、完全に設定します。つまり、**[検出の設定]** フォームで **[保存して閉じる]** をクリックします。(**[検出シード]** タブは、つながっているように見えますが、分離したフォームであり、データベースモデルの **[検出の設定]** フォームの一部ではありません。結果として、**[検出シード]** タブ上の情報を保存すると、NNMi はシード設定をすぐに更新します。)

検出の評価

ここでは、検出の進行状況と成功したかどうかを判定する方法を記載しています。

初期検出の進行状況をたどる

NNMi 検出は、動的かつ継続的です。完了することはないため、「検出完了」のメッセージが表示されることはありません。初回の検出と接続には、多少の時間がかかります。初期検出の進行状況を測定する方法を以下に示します。

- **[Network Node Manager i-series について]** ウィンドウで、予想レベルに達して安定化するノード数に注目します。このウィンドウは自動的に更新されません。初期検出中に、**[Network Node Manager i-series について]** ウィンドウを何回か開いてください。
- **[検出の設定]** フォームで、**[検出シード]** タブを見てください。このタブを、すべてのシードに「ノードが作成されました」結果が表示されるまで更新してください。「ノードが作成されました」結果は、デバイスがトポロジデータベースに追加されたことを示します。この結果は、NNMi がデバイスからすべての情報を収集してデバイスの接続を処理したことを示すものでは*ありません*。
- 代表ノードの **[ノード]** フォームを開きます。**[検出状態]** フィールドが「検出が完了」に移行したとき、NNMi はノードの基本的特性の他にノードの隣接 ARP キャッシュや隣接検出プロトコルも（該当する場合）収集を済ませています。この状態は、NNMi がデバイスの接続解析を完了したことを示す*ものではありません*。
- **[ノード]** インベントリビューで、ネットワークの様々な領域のキーデバイスが存在していることを確認します。
- 代表ノードの **[レイヤ 2 の近隣接続ビュー]** を開いて、その領域の接続解析が完了したかどうかを調べます。
- **[レイヤ 2 接続]** および **[VLAN]** インベントリビューを調べて、レイヤ 2 処理の進行状況を測定します。

すべてのシードが検出されているか？

1. **[検出の設定]** フォームを開きます。
2. **[検出シード]** タブで、ノードのリストを **[検出シードの結果]** 列でソートします。ノードがエラー状態の場合は、以下について検討してください。
   - ノードに到達できなかったか DNS 名が解決されなかった — ために検出が失敗したこれらのタイプの失敗に対しては、ノードへのネットワーク接続を確認して、DNS 名解決が正しいかどうかを調べてください。DNS 問題に対処するには、IP アドレスを使用してノードをシードするか、ホスト名を `hostNoLookup.conf` ファイルに加えます。
   - ライセンスノード数を超えた — この状況は、すでに検出されたデバイス数がライセンス限度に達したときに発生します。検出したノードをいくつか削除するか、ノードパックライセンスを追加購入します。
   - ノードが検出されたが SNMP 応答がない —SNMP 通信の問題は、シードされたデバイスだけでなく自動検出によって検出されたデバイスにも発生します。詳細については、通信の評価ページの 41 を参照してください。

すべてのノードには有効なデバイスプロファイルがあるか？

1 **[ノード]** インベントリビューを開きます。
2 **[デバイスプロファイル]** 列を、「デバイスプロファイルなし」文字列が含まれるようにフィルタリングします。
3 ノードが検出されてもデバイスプロファイルがない場合、**[設定] > [デバイスプロファイル]** ビューで新しいデバイスプロファイルを追加してから、ノード上で設定ポーリングを実行してデータを更新します。

すべてのノードが正しく検出されたか？

[ノード] インベントリビューでデータを調べます。管理アドレスがないノードがある場合は、これらのノードの通信設定をすべてのノードが SNMP 用に設定されましたか？ページの 42 の説明にしたがって確認します。

予想したノードが **[ノード]** インベントリビューにない場合は、以下について確認します。

- 見つからなかったノードごとに、検出プロトコル（たとえば CDP）が正しく設定されていることを確認します。
- 見つからないノードが WAN 上にある場合、そのノードを含む自動検出ルールの ping スィープを有効にします。

リストベース検出のみ

自動検出ルール

予期しない検出結果に遭遇した場合は、自動検出ルールを再検討します。

NNMi 検出でアドレスヒントが見つかる場合は、最初の一致ルールを使用してノードを作成するかどうかを判定しています。一致するルールがない場合、NNMi 検出はヒントを破棄します。自動検出ルールの順序番号によって、自動検出ルール設定が適用される順序が決まります。

それぞれの自動検出ルールで、以下の設定を確認してください。

- **[自動検出を有効にする]** を有効にして、ルールに対して自動検出が実行されるようにする必要があります。
- 以下の設定が、検出したいノードのタイプに対して正しいかどうかを確認します。
  - **SNMP デバイスを検出する**
  - **非 SNMP デバイスを検出する**

  デフォルトではルーターとスイッチのみが検出されて、SNMP 以外のノードは検出*されない*ことを忘れないでください。ご使用の環境を考慮せずにこれらの設定を有効にすると、NNMi が予期した以上のノードを検出してしまう可能性があります。

IP アドレス範囲

検出ヒントの IP アドレスは、IP アドレス範囲リスト内の **[ルールに含める]** エントリに一致する必要があります。含まれる IP アドレス範囲が自動検出ルールの中にない場合、すべてのアドレスヒントが一致とみなされます。（その場合は、自動検出ルールを設定する場合のヒントページの 53 を参照してください。）さらに、ヒントは **「ルールにより無視」** とマークされたエントリと一致*してはなりません*。すべてのチェックが正常に一致すると、このルールの設定がヒントの処理に使用されます。

- 予想したデバイスのいくつかが検出されない場合、設定した IP 範囲を確認してそのデバイスの IP アドレスが範囲の中に含まれていて小さい順序番号のルールで無視されないようにしてください。

- 必要以上のデバイスが検出されている場合は、含む範囲を変更するか、検出したくないデバイスの IP アドレスの無視される範囲を追加してください。また、**[SNMP デバイスを検出する]** が有効かも調べてください。

システムオブジェクト ID の範囲

検出ヒントのシステムオブジェクト ID (OID) は、システムオブジェクト ID 範囲リストの中の **[ルールに含める]** エントリと一致する必要があります。含まれるシステムオブジェクト ID 範囲が自動検出ルールの中にない場合、すべてのオブジェクト ID が一致とみなされます。さらに、OID は **「ルールにより無視」** とマークされたエントリと一致してはなりません。すべてのチェックが正常に一致すると、このルールの設定がヒントの処理に使用されます。

- システムオブジェクト ID 範囲を使用して、自動検出を拡大してデフォルトのルーターおよびスイッチ以外も含めるか、特定のルーターおよびスイッチを除外します。
- 各ノードは、検出されてトポロジデータベースに追加される前に指定された IP アドレス範囲とシステムオブジェクト ID 範囲の両方と一致する必要があります。

すべての接続と VLAN は正しいか？

NNMi はレイヤ 2 接続と VLAN を、デバイスがトポロジに追加された後の別個のステップとして作成します。NNMi に接続と VLAN を評価する前の初期検出として十分な時間を考慮してください。

レイヤ 2 の接続を評価するには、対象とする各ネットワーク領域のノードグループを作成し、続いてそのノードグループのトポロジマップを表示します。（**[ノードグループ]** インベントリで、ノードグループを選択して、**[アクション]** > **[ノードグループマップ]** をクリックします。）このマップで他のノードに接続していないノードを探します。

VLAN を評価するには、**[VLAN]** インベントリビューから、各々の **[VLAN]** フォームを開いて、その VLAN のポートのリストを調べます。

デバイスを再検出する

1. デバイスの設定ポーリングを実行します。
2. デバイスを削除します。

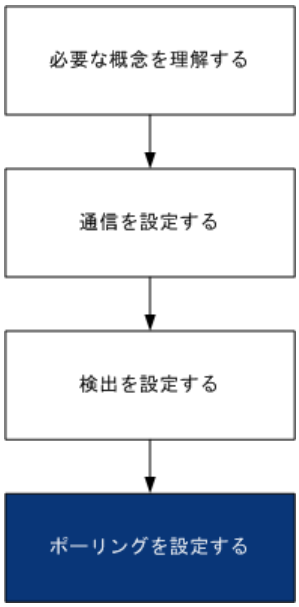
そのデバイスがシードの場合、シードを削除し、それからシードを再度追加します。

検出の調整

標準的な検出が行われるようにするためには、検出設定を調整して重大なデバイスと重要なデバイスのみが検出されるようにしてください。

- IP アドレス範囲またはシステムオブジェクト ID（またはその両方）でフィルタリングします。
- 非 SNMP デバイスと SNMP デバイス（スイッチでもルーターでもないデバイス）の検出を制限します。

NNMi 状態ポーリング



この章では、HP Network Node Manager i Software State Poller サービスを設定し、ネットワークモニタリングを拡張および微調整するのに役立つ情報を示します。この章は、NNMi ヘルプの情報を補充するものです。モニタリング動作方法の紹介、およびモニタリング設定方法の詳細は、NNMi ヘルプの「*ネットワークの稼働状態のモニタリング*」を参照してください。

NNM 6.x/7.x で作業した経験があり、NNMi 8.10 でモニタリングがどのように変更されたか理解したい場合は、相違点の高レベルの概要に関するステータスモニタリングページの 139 を参照してください。

この章では、以下の内容を記載しています。

- 状態ポーリングの概念
- 状態ポーリングの計画を作成
- 状態ポーリングの設定
- 状態ポーリングの評価
- 状態ポーリングの調整

状態ポーリングの概念

この項では、State Poller がポーリンググループの評価に使う順序など、ネットワークモニタリングの簡単な概要を示します。この項を読んだ後、さらに詳細な情報については状態ポーリングの計画を作成ページの 58 に進んでください。

ネットワーク検出と同じように、ネットワークでクリティカルであるか、または最も重要なデバイスのネットワークモニタリングに関心を集中する必要があります。NNMi では、トポロジデータベースでのみデバイスをポーリングできます。NNMi がどのネットワークデバイスをモニタリングするか、使用するポーリングの種類、およびポーリングする間隔を制御できます。

[モニタリングの設定] フォームのインタフェースとノードの設定を使って、デバイスのステータスポーリングを高度化し、さまざまなクラス、インタフェースの種類、およびノードの種類についてポーリングの種類と間隔を設定することができます。

State Poller のデータ収集が ICMP (ping) 応答を基礎にするように、または SNMP データを基礎にするように設定できます。NNMi は、ユーザーが有効にするデータ収集の種類から、実際の MIB オブジェクトへの内部的なマップを自動処理し、設定を大幅に簡単にします。

ポーリング設定の計画を作成するときは、State Poller サービス用にインタフェースグループとノードグループをセットアップする方法を注意深く考える必要があります。グループという概念が初めての場合は、その概要についてノードグループページの 29 とノード / インタフェース / アドレス階層ページの 33 を参照してください。

評価の順序

インタフェースまたはノードは複数のグループに属することがあるので、State Poller は、明確に定義された評価順序で、設定されたポーリング間隔およびポーリング種類を適用します。検出されたトポロジ内の各オブジェクトについて：

1. オブジェクトがインタフェースの場合、State Poller は基準を満たすインタフェースグループを探します。グループは最も小さい順序番号から最も大きい順序番号へという順序で評価されます。最初に一致するグループが使われ、その時点で評価は停止します。
2. オブジェクトを把握したインタフェースグループがない場合、グループは最も小さい順序番号から最も大きい順序番号への順序で評価されます。最初に一致するグループが使われ、その時点で評価は停止します。含まれているインタフェースのうち、独自の特性に関してインタフェースグループの基準を満たしていないものは、ホストであるノードからポーリング設定を継承します。
3. 検出されたものの、ノードまたはインタフェースの設定定義に含まれないデバイスは、グローバルなモニタリング設定（**[モニタリングの設定]** フォームの **[デフォルト設定]** タブ）によってモニタリング動作が確定されます。

状態ポーリングの計画を作成

この項では、ポーリング設定チェックリストなど、State Poller 設定の計画作成について説明します。モニタリングの計画作成に便利な詳細情報によって、ポーリンググループの作成法が決まり、ポーリングプロセスの間にどの種類のデータを取得する必要があるかが決まります。

ポーリングチェックリスト

次のチェックリストを使って、State Poller 設定の計画を作成できます。

- ☐ モニタリングできるものは何でしょう？
- ☐ オブジェクトの種類、場所、相対的重要性、その他の基準に基づいて、モニタリングされるアイテムの論理グループは何であるか？
- ☐ 各グループはどのくらいの頻度でモニタリングする必要があるか？
- ☐ モニタリングされるアイテムの情報を取得するために、何のデータを収集する必要があるか？ 次のものが含まれることがあります。
  - — ICMP (ping) 応答
  - — SNMP 障害データ
  - — NNM iSPI for Performance ライセンスがある場合は、SNMP パフォーマンスデータ
  - — 追加の SNMP コンポーネント稼働状態データ

ポーリング設定の例

ポーリング設定プロセスの理解を深めるために、次の例について考えます。ネットワークに ProximiT の最新のプロキシサーバーが含まれていると仮定します。これらのデバイスに到達できることを確認する必要がありますが、プロキシサーバーの SNMP モニタリングは要求しません。

1. モニタリングできるものは何でしょう？

   検出されたものだけをモニタリングできるので、自動検出ルールを設定して、NNMi のデータベースに自分の ProximiT プロキシサーバーが a あることを確認します。検出の設定の詳細は、NNMi 検出ページの 45 を参照してください。

2. モニタリングされるアイテムの論理グループは何でしょうか？

   明らかに、ProximiT プロキシサーバーを一緒にグループにまとめ、同じモニタリング設定をこれらすべてに適用したいと考えています。デバイスのインタフェース (SNMP) モニタリングを行っているのですから、インタフェースグループは必要ありません。

   このノードグループを使って、ビューをフィルタし、プロキシサーバーのステータスをグループとしてチェックし、グループをサービス外にしてファームウェアを更新することもできます。

3. 各グループはどのくらいの頻度でモニタリングする必要があるでしょうか？

   サービスレベル契約条項で、プロキシサーバーについて 5 分間のポーリング間隔で十分です。

4. どのデータを収集する必要があるでしょうか？

   モニタリング設定が他のグループと異なるのは次の点です。ProximiT proxy サーバーの例として、ICMP 障害のモニタリングを有効にし、SNMP 障害およびポーリングのモニタリングを無効にします。グループについての SNMP 障害モニタリングがない場合、コンポーネント稼働状態モニタリングは適用されません。

これらの設定選択肢に関する計画作成情報の詳細は、以下の項を参照してください。

— 何をモニタリングできるでしょうか？ページの 59

— グループの計画作成ページの 61

— ポーリング間隔の計画作成ページの 63

— どのデータを収集するかの決定ページの 64

何をモニタリングできるでしょうか？

デフォルトで、NNMi State Poller は SNMP ポーリングを使って以下をモニタリングします。

- NNMi 検出対象デバイス上で既知の別のインタフェースに接続されたインタフェース。
- IP アドレスをホストするルーターインタフェース。

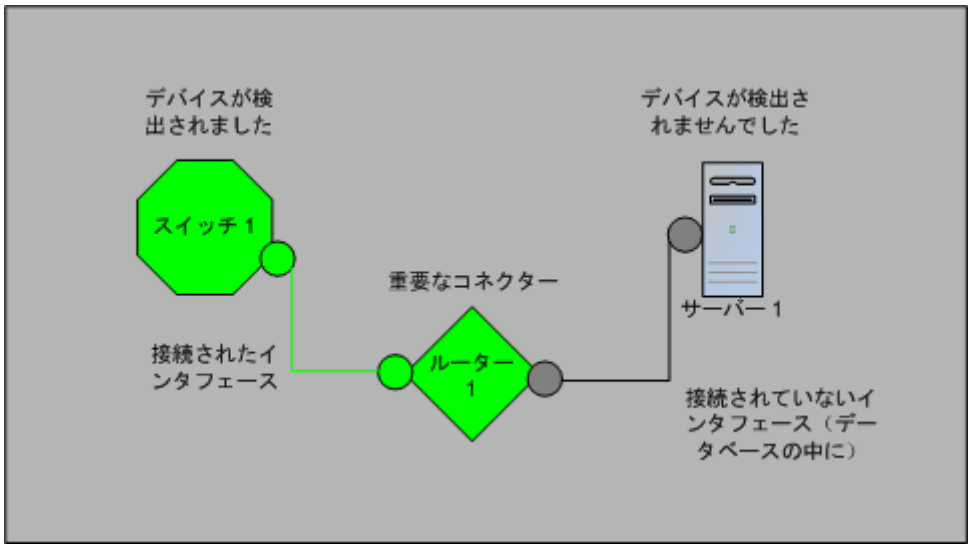
ほとんどの場合、インタフェースに接続されたポーリングによってのみ、十分に正確な根本原因分析ができます。モニタリング対象インタフェースのセットを拡張すると、ポーリングのパフォーマンスに影響が及ぶ可能性があります。

モニタリングの拡張

モニタリングを拡張して、以下が含まれるようにできます。

- 未接続インタフェース。デフォルトでは、NNMi がモニタリングする未接続インタフェースは IP アドレスのあるもののみであり、**ルーター**ノードグループに含まれます。

NNMi は、次のように、NNMi が検出した別のデバイスに接続されていないインタフェースとして未接続インタフェースを定義します。



- ルーターインタフェースのように、IP アドレスのあるインタフェース。
- SNMP をサポートしないデバイス用の ICMP ポーリング。デフォルトで、ICMP ポーリングは、**非 SNMP デバイス**ノードグループについて有効です。

モニタリングされないノードへのインタフェース

直接管理していないデバイスに接続されているインタフェースのステータスを知る必要があることがあります。たとえば、アプリケーションまたはインターネットサーバーへの接続が確立されているかどうか知る必要があるものの、そのサーバーのメンテナンスは担当していないことがあります。検出ルールにそのサーバーを組み入れていないと、NNMi はそのサーバーに面するインタフェースを未接続と見なします。

モニタリングされていないノードに接続する重要なインタフェースのステータスをモニタリングする方法には次の 2 つがあります。

- モニタリングされていないノードの検出。

  モニタリングされていないノードを NNMi トポロジに追加するとき、NNMi は、トポロジの残りの部分にノードを接続しているインタフェースを接続済みと見なします。この場合、NNMi は、モニタリング設定に従ってこれらのインタフェースをポーリングできます。NNMi はノードを管理対象として検出します。NNMi にモニタリングさせたくない、管理されていないノード。

検出された各ノードは、NNMi が積極的にそのノードを管理しているかどうかに無関係に、ライセンスの最大数まで数えられます。

- 未接続インタフェースのポーリング

  未検出ノードの接続を備えたネットワークデバイスを含むノードグループを作成できます。次に、ノードグループの未接続インタフェースのポーリングを有効にします。

  NNMi は、多数のインタフェースのあるデバイスに大量のトラフィックを追加できる、ノードグループのデバイス上のインタフェースをすべてポーリングします。

モニタリングの停止

NNMi 管理モードを使って、デバイスまたはインタフェースを管理対象外またはサービス外に設定できます。管理対象外は恒久的な状況と見なされます。オブジェクトのステータスを知る心配をする必要はありません。サービス外は一時的な状況と見なされます。オブジェクトはオフラインになり、ダウンインシデントは過剰になります。

すべてのグループ設定全体のオーバーレイとして、管理モードを考えてください。グループ、ポーリング間隔、種類に無関係に、オブジェクトのステータスが管理対象外またはサービス外に設定されている場合、State Poller はそのオブジェクトと通信しません。

ベストプラクティス

検出を行い、データベースに配置することを選択したデバイスまたはインタフェースの中には、ポーリングの必要がないものもあります。管理対象外に恒久的に設定するオブジェクトに注意してください。1 つまたは複数のノードグループを作成し、管理モードをもっと簡単に設定することもできます。

グループの計画作成

ノードグループとインタフェースグループをセットアップしてから、モニタリングを設定する必要があります。したがって、ノードグループとインタフェースグループを設定するときはポーリング要求について考慮する必要があります。重要なデバイスを頻繁にモニタリングできるようにノードグループとインタフェースグループを設定するのが理想的です。クリティカルでないデバイスのチェックをあまり頻繁でないようにできます（そもそもチェックを行う場合です）。

これらグループは、**[設定] > [ノードグループ]** または **[設定] > [インタフェースグループ]** サークスペースを使用して定義します。これらグループは、デフォルトで、インシデント、ノード、インタフェース、およびアドレスビューをフィルタするのに使うのと同じグループです。モニタリング設定用にノードフィルタまたはインタフェースフィルタの別個のセットを作成するには、ノードグループまたはインタフェースグループを開き、**[ノードグループ]** フォームまたは **[インタフェースグループ]** フォームで **[ビューフィルタリストに追加]** チェックボックスをオンにします。**[保存して閉じる]** をクリックします。

[モニタリングの設定] フォームの **[ノードの設定]** タブと **[インタフェース設定]** タブにあるノードグループまたはインタフェースグループのレベルで、ポーリングの種類とポーリングの間隔を設定します。

類似のポーリングのニーズごとに、インタフェースまたはデバイスをグループにまとめる基準を決定します。計画作成に際して考慮すべきいくつかの要因は次のとおりです。

- ネットワークのどのエリアにこれらのデバイスがあるか？タイミング制限があるか？
- デバイスの種類ごとに収集したポーリング間隔またはデータを差別化するか？インタフェースの種類ごとにか？
- NNMi には使用できる事前設定されたグループがあるか？

ベストプラクティス

同時にサービス外になりそうなオブジェクトのグループ定義を、場所ごとであれ、他の何らかの基準ごとであれ、作成することができます。たとえば、IOS アップグレードを適用しながら、すべての Cisco ルーターをサービス外モードにできます。

インタフェースグループ

基準に基づいて、どのインタフェースグループを作成するか決定します。インタフェースグループが最初に評価されることを覚えておいてください（状態ポーリングの概念ページの 57参照）。インタフェース グループはノード グループ メンバーシップを参照できるので、計画を実現するインタフェース グループの前に、ノード グループの設定を完了できます。

事前設定されたインタフェース グループ

NNMi には、使用できるように既に設定済みの便利なインタフェース グループがいくつかあります。たとえば、次のとおりです。

- ISDN 接続に関連付けられた IFType のある全インタフェース
- 音声接続用のインタフェース
- ポイントツーポイント通信用のインタフェース
- ソフトウェアループバックインタフェース
- VLAN インタフェース
- リンク集合プロトコルに関与するインタフェース

HP は、時間をかけてさらに多くのデフォルトのグループを追加し、設定作業を簡単にして行きます。既存のグループを使用または変更するか、または自分専用のグループを作成できます。

インタフェースには次の 2 つの種類の修飾子があります。つまり、ホストであるノードと IFType のノードグループメンバーシップ、またはインタフェース用の他の属性です。これらは次のように組み合わせできます。

- ノードグループ内のノードの全インタフェースを IFType と無関係にグループにまとめます。IFType または属性（名前、エイリアス、説明、速度、インデックス、アドレスなど）は選択しません。
- 特定の IFType または属性のセットのインタフェースは、それらインタフェースが存在するノードに無関係にすべてグループにまとめられます。
- 特定のノードグループに存在する特定の IFType または属性のインタフェースのみがグループにまとめられます。

ノードグループ

インタフェースグループの計画を作成してから、ノードグループの計画を作成します。モニタリング用に作成された全ノードグループがフィルタビューに意味があるとは限らないので、ノードグループは独立に設定できます。

事前設定されたノードグループ

HP は、ノードグループのデフォルト集合を用意して、設定作業を簡単にしています。これらの基礎になっているのは、検出プロセスの間にシステムオブジェクト ID から導出されたデバイスカテゴリです。デフォルトのノードグループには以下が含まれます。

- ルーター
- ネットワーキングインフラストラクチャデバイス（スイッチ、ルーターなど）
- Microsoft Windows システム
- SNMP コミュニティ文字列を持っていないデバイス
- 重要ノード。Causal Engine によって内部的に使用されており、コネクタ障害の危険にさらされているデバイスの特殊処理を提供します。詳細は、NNMi ヘルプの「*事前定義されたビューフィルタとしてのノードグループ*」を参照してください。

HP は、時間をかけてさらに多くのデフォルトのグループを追加し、設定作業を簡単にして行きます。既存のグループを使用または変更するか、または自分専用のグループを作成できます。

次のノード属性を使用して、関連するノードの定義に条件を付けることができます。

- ノード上の IP アドレス
- ホスト名ワイルドカード規約
- デバイスプロファイル派生物。たとえば、カテゴリ、ベンダー、ファミリ
- MIB II sysName、sysContact、sysLocation

ベストプラクティス

簡単で再使用可能な極小のグループを作成し、モニタリングまたは視覚化のためにこれらを結合して階層クラスタにすることができます。グループ定義は重なり合います。たとえば、「すべてのルーター」と「IP アドレスの末尾が 100 のすべてのシステム」です。ノードは複数のグループに属することができると考えられます。

バランスを取るためには、使われない余分なエントリのリストで負担を大きくしないように、設定および表示用に豊富なグループのセットを作成します。

デバイスプロファイルとの相互作用

各デバイスが検出されると、NNMi はシステムオブジェクト ID を使用して、使用可能なデバイスプロファイルのリストにインデックスを作成します。デバイスプロパティは、ベンダー、製品、ファミリ、デバイスカテゴリなど、デバイスの追加属性を導出するために使用されます。

ノードグループを設定するとき、これら導出された属性を使用して、モニタリング設定に適用するデバイスをカテゴリにまとめられます。たとえば、ベンダーに無関係に、ネットワーク全体のすべてのスイッチを特定のポーリング間隔でポーリングすることもできます。デバイスカテゴリ「スイッチ」を自分のノードグループの定義特性として使えます。システムオブジェクト ID がカテゴリ「スイッチ」にマップされる、検出されたデバイスはすべて、ノードグループについての設定を受け取ります。

ポーリング間隔の計画作成

オブジェクトグループごとに、NNMi がデータを収集するのに使うポーリング間隔を選択します。サービスレベル契約条項に最も適切に一致するように、間隔は 1 分間と短くすることもできますし、数日間と長くすることもできます。

ベストプラクティス

間隔が短いと、可能な限り迅速にネットワーク問題を認識するのに役立ちます。しかし、あまりに短い間隔であまりに多くのオブジェクトをポーリングすると、State Poller にバックログを発生させる可能性があります。各自の環境について、リソース利用と間隔の間で最良のバランスを見つけてください。

どのデータを収集するかの決定

State Poller サービスは、ポーリングを使って、ネットワークでモニタリングされているデバイスに関する状態情報を収集します。ポーリングは ICMP または SNMP を使って実行できます。

ICMP (ping)

ICMP アドレスモニタリングは、ping 要求を使って、管理対象の各 IP アドレスの使用可能性を確認します。

SNMP

SNMP モニタリングは、モニタリングされている各 SNMP エージェントが SNMP クエリーに応答していることを確認します。

- State Poller は、間隔ごとに 1 つのクエリーで、モニタリングされている各オブジェクトから設定済み SNMP 情報を収集するよう、高度に最適化されています。設定の変更を保存すると、State Poller は、各オブジェクトのグループメンバーシップを再計算し、収集する設定済み間隔とデータセットに再適用します。
- SNMP モニタリングは、モニタリングされているすべてのインタフェースとコンポーネントに SNMP クエリーを発行し、MIB II インタフェーステーブル、HostResources MIB、およびベンダー特有の MIB から現在の値を要求します。障害モニタリングに使われる値もあります。NNM iSPI for Performance をインストールしてある場合は、パフォーマンス測定に使われる値もあります。

SNMP コンポーネント稼働状態データ

コンポーネント稼働状態をグローバルなレベルで有効または無効にできます。障害に関するコンポーネント稼働モニタリングは、デバイスの障害ポーリング間隔設定に従います。

ポーリングごとに追加データを収集しても、ポーリングを実行する時刻への影響はありません。しかし、各オブジェクトについて保存された追加データによって、State Poller 用にメモリ要求が増加する可能性があります。

➤ パフォーマンスモニタリング設定は NNM iSPI for Performance でのみ使用されます。パフォーマンスに関するコンポーネント稼働モニタリングは、デバイスのパフォーマンスポーリング間隔設定に従います。

ベストプラクティス

モニタリング設定変更をバッチ処理すると、State Poller の進行中の操作が混乱することは少なくなります。

状態ポーリングの設定

この項では、設定のヒントを示し、設定例をいくつか挙げます。この項を読んだ後、特定の手順については、NNMi ヘルプの「*モニタリング動作の設定*」を参照してください。

➤ 大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。詳細については、ベストプラクティス : 既存の設定を保存ページの 28 を参照してください。

インタフェースグループとノードグループの設定

[設定] ワークスペースでインタフェースグループとノードグループを作成します。詳細は、NNMi ヘルプの「*ノードまたはインタフェースのグループの作成*」を参照してください。

例　たとえば、ProximiT プロキシサーバー用にノードグループを設定する方法は次のとおりです。

1. **[設定] > [ノードグループ]** を開き、**[新規作成]** をクリックします。
2. グループ **`Proxy Servers`** という名前を挙げ、**[ビューフィルタリストに追加]** をオンにします。
3. **[ホスト名ワイルドカード]** タブを選択し、**[新規作成]** をクリックします。
4. ワイルドカードを **`prox*.yourdomain.com`** として入力し、**[保存して閉じる]** をクリックします。

   ProximiT デバイスについて Device Profile（デバイスプロファイル）と Category（カテゴリ）を設定してある場合は、**[デバイスフィルタ]** タブを使って **[デバイスカテゴリ]** セレクタにアクセスし、作成した Proxy Server カテゴリをグループの基礎にすることができます。
5. グループ定義で **[保存して閉じる]** をクリックします。

ノードグループを設定してから、インタフェースグループ設定でノードグループを参照する必要があります。

インタフェースモニタリングの設定

State Poller は、ノードグループの前に、インタフェースグループメンバーシップを分析します。作成した各インタフェースグループ、および使用する既存のインタフェースグループごとに、**[モニタリングの設定]** ダイアログと **[インタフェース設定]** タブを開き、State Poller がそのグループを処理する方法に関する指示のカスタムセットを作成します。指示には以下のものが含まれます。

- 障害ポーリングの有効化または無効化
- 障害ポーリング間隔の設定
- NNM iSPI for Performance がある場合、パフォーマンスポーリングの有効化または無効化
- NNM iSPI for Performance がある場合、パフォーマンスポーリング間隔の設定
- NNM iSPI for Performance がある場合、パフォーマンス管理しきい値の設定
- グループ内の未接続インタフェース（または IP アドレスをホストしている未接続インタフェース）をモニタリングするかどうかの選択

インタフェースグループごとに異なる設定ができます。State Poller は、小さい順序番号から大きい順序番号へとリストを評価することを覚えておいてください。

ベストプラクティス　複数のグループにあてはまるオブジェクトは最も順序番号の小さいグループから設定を適用されることを頭に入れておきつつ、順序番号をダブルチェックします。

ノードモニタリングの設定

あるオブジェクトが設定済みのインタフェースグループにあてはまらない場合、State Poller はノードグループ内のメンバーシップについて、そのオブジェクトを評価します。最も小さい順序番号から最も高い順序番号へと、設定は最初の合致するノードグループに適用されます。

ノードグループごとに、［**モニタリングの設定**］フォームを開いてから［**ノードの設定**］タブを開きます。State Poller がグループを処理する方法に関する指示のカスタムセットを作成します。指示には以下のものを入れられます。

- 障害ポーリングの有効化または無効化
- 障害ポーリング間隔の設定
- NNM iSPI for Performance がある場合、パフォーマンスポーリングの有効化または無効化
- NNM iSPI for Performance がある場合、パフォーマンスポーリング間隔の設定
- NNM iSPI for Performance がある場合、パフォーマンス管理しきい値の設定
- グループ内の未接続インタフェース（または IP アドレスをホストしている未接続インタフェース）をモニタリングするかどうかの選択

ノードグループごとに異なる設定ができます。

ベストプラクティス　複数のグループにあてはまるオブジェクトは最も順序番号の小さいグループから設定を適用されることを頭に入れておきつつ、順序番号をダブルチェックします。

デフォルト設定の確認

State Poller は、定義済みのインタフェース設定またはノードの設定に合致しないオブジェクトについて［**デフォルト設定**］タブの設定を適用します。このタブの設定を検討し、デフォルトレベルで自分の環境に合致することを確認します。たとえば、デフォルト設定としてすべての未接続インタフェースをポーリングすることはほとんどありません。

➤ 変更を実現するためには、コンソールに戻り、すべての［**設定のモニタリング**］ダイアログボックスを必ず［**保存して閉じる**］ようにしてください。

状態ポーリングの評価

この項では、モニタリング設定の進行と成功を評価する方法をリストします。

ネットワークモニタリングの設定を確認します

NNMi が指定のノードまたはインタフェースのモニタリングに使う設定を決定すると、ステータスポーリングをいつでも開始できます。

インタフェースまたはノードは正しいグループのメンバーでしょうか？

あるグループにどのインタフェースまたはノードが属するか確認するには、［**設定**］ワークスペースで次の 1 つを選択します。

- ノードグループ
- インタフェースグループ

ヘルプの指示に従って、グループのメンバーを表示します。オブジェクトは複数のグループのメンバーになれること、他のグループの順序番号の方が小さい可能性があることを頭に入れておいてください。

その代わりに、オブジェクト（インタフェースまたはノード）を開き［**グループ**］タブをクリックして、オブジェクトが属するグループの完全なリストを表示することもできます。このリストは、グループ名ごとにアルファベット順であって、どの設定が適用されるかを決定する順序番号を反映していません。

オブジェクトがグループのメンバーでない場合は次のとおりです。

1 ［インベントリ］ビューのデバイスプロファイルを取得します。

2 **［設定］＞［デバイスプロファイル］**下にあるデバイスプロファイルに関する属性マップを確認します。

3 ノードグループ定義の属性要件を確認します。

不一致がある場合は、［デバイスプロファイル］に由来するカテゴリを調整して、その種類のデバイスが自分のノードグループに当てはまるようにできます。**［アクション］＞［ポーリングの設定］**を実行して、ノードが当てはまるようにノードの属性を更新する必要があるかもしれません。

どの設定が適用されていますか？

特定のノード、インタフェース、またはアドレスに有効なモニタリング設定をチェックするには、該当する［インベントリ］ビュー内のそのオブジェクトを選択し、**［アクション］＞［モニタリング設定］**を選択します。NNMi に現在のモニタリング設定が表示されます。

［有効化された障害ポーリング］と**［障害ポーリング間隔］**の値を調査します。これらの値が予想どおりでない場合は、**［ノードグループ］**または**［インタフェースグループ］**の値を見て、どの順序付けられたグループ一致が適用されるか調べます。

オブジェクト用にトラフィックが無効にされていないことを確認するために、オブジェクトの**［アクション］＞［通信設定］**をチェックする必要があることもあります。

どのデータが収集されていますか？

特定のデバイスのステータスポーリングを開始し、予想された種類のポーリング (SNMP、ICMP) がそのデバイスについて実行されていることを確認できます。ノードを選択し、**［アクション］＞［ステータスポーリング］**をクリックします。NNMi はデバイスのリアルタイムのステータスチェックを実行します。実行中のポーリングの種類と結果は出力に表示されます。ポーリングの種類が予想したものでない場合は、ノードのモニタリング設定、およびモニタリング設定のそれぞれのグローバル、インタフェース、またはノードに関する設定をチェックします。

ステータスポーリングのパフォーマンスの評価

自分の環境のステータスポーリングのパフォーマンスを評価するには、State Poller 稼働状態チェックの情報を使って、State Poller サービスの動作を数値で表し、評価します。

State Poller は最新の状態に付いていっていますか？

［ヘルプ］>［**Network Node Manager i-series について**］で State Poller サービスの現在の稼働状態統計をいつでもチェックできます。表 2 は、［**HP Network Node Manager i-series について**］ウィンドウに NNMi が表示する State Poller 稼働状態情報をリストします。

表 2　State Poller 稼働状態情報

情報	説明
ステータス	State Poller サービスの全般的なステータス
ポーリングカウンタ	• 最後の 1 分に要求された収集 • 最後の 1 分に完了された収集 • 進行中の収集
最後の 1 分にスキップを実行する時刻	設定済みのポーリング間隔内で完了しなかった、定期的にスケジュールされたポーリングの数。値がゼロでない場合は、ポーリングエンジンが最新の状態に付いていっていないか、またはターゲットが応答より速くポーリングされています。 • 監視すべきもの：この値が増加し続ける場合は、ターゲットとの通信に問題があるか、または NNMi の負荷が過剰です。
古い収集	古い収集というのは、少なくとも 10 分間、ポーリングエンジンから応答を受信していない収集のことです。稼働状態が良好なシステムでは古い収集はありません。 • 監視すべきもの：この値が一定して増加する場合は、ポーリングエンジンに問題があります。
ポーリング結果のキューの長さ	• 監視すべきもの：この値はほとんどの時間 0 に近いはずです。 • 実行すべきアクション：キューのサイズがきわめて大きい場合、OVJBoss はメモリを越えて実行されている可能性があります。
State Mapper のキューの長さ	• 監視すべきもの：この値はほとんどの時間 0 に近いはずです。 • 実行すべきアクション：このキューのサイズがきわめて大きい場合は、NNMi システムと NNMi データベースのパフォーマンスをチェックします。

状態ポーリングの調整

状態ポーリングのパフォーマンスは次の重要な変数の影響を受けます。

- ポーリングされるデバイス / インタフェースの数
- 設定されるポーリングの種類
- 各デバイスのポーリングの頻度

これらの変数は、ネットワーク管理のニーズによって促進されます。ステータスポーリングについてパフォーマンス上の問題がある場合は、次の設定を考慮してください。

- 個別のノードのポーリング設定はノードグループとインタフェースグループ内のメンバーシップによって制御されるので、類似のポーリング要求のあるノードまたはインタフェースがグループに含まれていることを確認します。
- 未接続インタフェースまたは IP アドレスをホストするインタフェースをポーリングしている場合は、設定をチェックして、必要なインタフェースのみをポーリングしていることを確認します。**［ノードの設定］**フォームまたは**［インタフェース設定］**フォーム（**［モニタリングの設定］**フォームでグローバルにではなく）でこれらのポーリングを有効にし、最も特定な制御を維持し、ポーリングするインタフェースの最も小さいサブセットを選択します。
- 未接続インタフェースのポーリングでは、未接続の*すべて*のインタフェースがモニタリングされることを覚えておいてください。IP アドレスのある未接続のインタフェースのみをモニタリングするには、IP アドレスをホストするインタフェースのポーリングを有効にします。

モニタリング設定とは無関係に、ステータスポーリングは、ネットワーク応答性に左右され、全般的なシステムパフォーマンスの影響を受ける可能性があります。デフォルトのポーリング間隔のあるステータスポーリングは多くのネットワーク負荷をかけませんが、サーバーとポーリングされているデバイスの間のネットワークリンクのパフォーマンスが低い場合、ステータスポーリングのパフォーマンスも低くなる可能性があります。タイムアウトを大きく、再試行の数を小さく設定すると、ネットワーク負荷を低減できますが、これらの設定変更でできるのはそれだけです。タイミングの良いポーリングを行うには、適切なネットワークパフォーマンスと十分なシステムリソース（CPU、メモリー）が必要です。

コンポーネント稼働状態モニタリングを有効または無効にしても、ポーリングのタイミングには影響がありません。スケジュールされた時刻に、追加の MIB オブジェクトが収集されるだけです。しかし、コンポーネント稼働状態モニタリングを無効にすると、State Poller が使用するメモリーの量が減少する可能性があります。

管理

この項では以下の章について説明します。

- https の有効化（対象：NNMi）
- 高可用性クラスタに HP NNM i-series Software を設定する
- NNMi のバックアップおよびリストアツール
- NNMi 管理サーバーの変更

https の有効化（対象：NNMi）

デフォルトで、HP Network Node Manager i Software Web サーバーは、データの NNMi コンソール への転送に http プロトコルを使用します。これは NNMi コンソール のすべてのユーザーがイントラネット内にいる環境には適切です。イントラネット外のユーザーが NNMi コンソールにアクセスする必要がある場合は、https プロトコルを有効にすると、NNMi Web サーバーと Web ブラウザの間を移動するデータを暗号化できます。

安全なソケットレイヤー (SSL) テクノロジの詳細およびこのトピックでの設定手順については、以下にアクセスしてください。

http://tomcat.apache.org/tomcat-6.0-doc/ssl-howto.html

NNMi Web サーバー用に https プロトコルを有効にするには、以下の手順を完了します。

- パブリックキー証明書の入手とインストール
- server.xml ファイルの更新
- 新しい接続情報をユーザーに通信

パブリックキー証明書の入手とインストール

パブリックキー証明書は Web サーバーをユーザーに識別します。この証明書は、自己署名または第三者署名にできます。

自己署名証明書

どのシステム管理者も 1 つの Web サーバー用に自己署名証明書を作成できます。自己署名証明書が便利なのは、Web サーバーのすべてのユーザーが Web アプリケーション（この場合は NNMi コンソール）を備えた組織と既に信頼関係を持っている場合です。

自己署名証明書を入手し、インストールするには、下記の手順に従ってください。

1 Java keytool ジェネレータを使用して証明書ファイルを作成します。

- このコマンドラインツールを使用するには、次の例に類似したコマンドを実行します。

```
keytool -genkey -keystore "c:\path\to\my\.keystore"
-storepass changeit -keypass changeit -keyalg RSA
-validity 3650 -alias alias
```

- 対話形式のツールを使用するには、次のコマンドを実行します。

```
keytool -genkey
```

2 keytool コマンドを実行している間に指定した場所に証明書ファイルを移動します。

第三者証明書

第三者証明書は、証明書の購入者のみが証明書を使用していることを確定できるビジネスが作成および保守します。このビジネス関係は、Web サーバーを有効な存在が提供しているという信頼を形成します。各自の組織に既に第三者証明書がある場合は、この証明書を NNMi Web サーバーに使用できます。

第三者証明書を入手およびインストールするには、この章で前述した Tomcat 設定ヘルプの指示に従います。

server.xml ファイルの更新

NNMi 8.10 とともに配布される jboss のバージョンには Tomcat サーバーバージョン 6.0 が含まれています。Tomcat 6.0 用の https 設定プロセスは Tomcat の以前のバージョンのプロセスとは異なります。

Tomcat サーバーは次のファイル内に設定されます。

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

$jboss.home.dir のデフォルト値は次のとおりです。

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

jboss.http.port 変数と jboss.https.port 変数は、それぞれ http または https プロトコルを識別するために、サーバー設定コネクタブロック内で使用されます。これらの変数は次のファイルに定義されます。

- *Windows*: %NnmDataDir%\shared\nnm\conf\nnm.ports.properties
- *UNIX*: $NnmDataDir/shared/nnm/conf/nnm.ports.properties

コンピュータで既に使用中のポートを識別するには、netstat -a コマンドを使用します。

server.xml ファイルのコネクタブロックは、Web サーバーと Web ブラウザの間の接続パラメータを指定します。server.xml ファイルには複数のコネクタブロックを入れることができます。

安全でない接続のデフォルトコネクタブロックは、server.xml ファイルの <!-- HTTP CONFIG --> セクションにあり、次の例に類似しています。

```
<Connector port="${jboss.http.port}"
    address="${jboss.bind.address}"
    maxThreads="250"
    maxHttpHeaderSize="8192"
    emptySessionPath="true"
    protocol="HTTP/1.1"
    enableLookups="false"
    redirectPort="${jboss.https.port:8443}"
    acceptCount="100"
    connectionTimeout="20000"
    disableUploadTimeout="true"/>
```

イントラネット内部のユーザーを http プロトコルで NNMi コンソールと接続させる場合は、このデフォルトブロックを所定の位置のままにしておきます。すべてのユーザーを https プロトコルで NNMi コンソールと接続させる場合は、このブロックから削除するか、またはコメントアウトします。

https プロトコルを指定するには、該当する https コネクタブロックをコメントでないようにし、該当するように変更します。

server.xml ファイルを変更した後で、NNMi サービスを再起動します。

```
ovstop
ovstart
```

- 自己署名証明書を使用している場合は、server.xml ファイルの <!-- HTTPS/SSL CONFIG --> セクションの最初のコネクタブロックを使用します。これは次の例に類似しています。このブロックでは、太線は自己署名証明書を識別します。

```
<Connector port="${jboss.http.port}"
    address="${jboss.bind.address}"
    maxThreads="250"
    acceptCount="100"
    connectionTimeout="20000"
    strategy="ms"
    maxHttpHeaderSize="8192"
    emptySessionPath="true"
    scheme="https"
    secure="true"
    SSLEnabled="true"
    keystoreFile="${path-to-file}\.keystore-filename"
    keystorePass="changeit"
    keyAlias="alias"
    clientAuth="false"
    sslProtocol="TLS" />
```

- 第三者証明書を使用している場合は、server.xml ファイルの <!-- HTTPS/SSL CONFIG --> セクションの 2 番目のコネクタブロックを使用します。これは次の例に類似しています。このブロックでは、太線は第三者証明書を識別します。

```
<Connector port="${jboss.http.port}"
    address="${jboss.bind.address}"
    maxThreads="250"
    acceptCount="100"
    connectionTimeout="20000"
    strategy="ms"
    maxHttpHeaderSize="8192"
    emptySessionPath="true"
    scheme="https"
    secure="true"
    SSLEnabled="true"
    SSLCertificateFile="/usr/local/ssl/server.crt"
    SSLCertificateKeyFile="/usr/local/ssl/server.pem"
    clientAuth="false"
    sslProtocol="TLS" />
```

コネクタブロック内のパラメータの詳細は、この章で前述した Tomcat 設定ヘルプを参照してください。

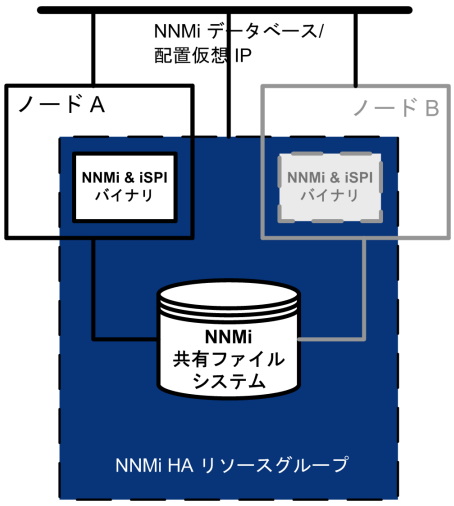
新しい接続情報をユーザーに通信

イントラネット外部のユーザーの場合、NNMi コンソールにアクセスするための新しい URL は次のとおりです。

https://*<host_name>*:*<https_port>*/nnm/

デフォルトの https ポート (443) を使用している場合、ユーザーはポート番号を URL に入れる必要はありません。

高可用性クラスタに HP NNM i-series Software を設定する



高可用性 (HA) とは、構成された動作中のハードウェアおよびソフトウェアの一部に障害が発生しても中断されないサービスを提供するシステムです。HA クラスタは、フェイルオーバー発生時の機能とデータの継続性を保証するために、協調して動作するハードウェアとソフトウェアのグループ化を定義します。

NNMi では、HP Network Node Manager i Software バージョン 8.02 から、別途購入が必要な HA 製品を使って構成される HA クラスタ内で NNMi を実行する設定をサポートするようになりました。NNMi バージョン 8.10 では、NNMi Smart Plug-in (iSPI) も、HA 下で実行できるようになります。

この章では、HA 環境で NNMi を設定するためのテンプレートと iSPI について説明します。ここでは、HA 製品の詳細な設定手順については説明しません。NNMi に用意されている HA 設定コマンドは、サポートされる HA 製品用のコマンドに関するラッパーとなります。HA 製品固有のコマンドの代わりに、以下の手順で説明している NNMi のコマンドを使用できます。

この章では、以下の内容を記載しています。

- サポート対象の HA 製品
- HA 用に NNMi を設定するための前提条件
- HA の概念
- HA の設定
- 共有 NNMi データ
- HA クラスタ内の NNMi のライセンス契約
- HA 設定のメンテナンス
- HA の設定解除
- HA 下の NNMi を NNMi 8.0x から NNMi 8.10 にアップグレードする
- HA 設定のトラブルシューティング
- HA 設定リファレンス

サポート対象の HA 製品

HA 下で NNMi を設定 / 実行するために使う、NNMi に付属しているコマンドは、各種のオペレーティングシステム用の以下の HA 製品で動作します。

- *HP-UX* の場合：HP ServiceGuard バージョン 11.18 以降
- *Linux* の場合：HP ServiceGuard バージョン 11.18 以降
- *Solaris* の場合：Veritas Cluster Server バージョン 5.0
- *Windows* の場合：Microsoft Cluster Services for Windows 2003

この章で説明するテンプレートを使って、NNMi を他の HA 製品下で実行するように設定できますが、その設定に関する、クラスタ設定問題のサポートは行いません。

HA 用に NNMi を設定するための前提条件

特定のシステムを NNMi HA クラスタのノードとして追加するには、そのシステムは以下の必要条件を満たしている必要があります。

- 仮想 IP アドレス使用のサポート
- 共有ディスク使用のサポート
- 『*HP NNMi ソフトウェアシステムとデバイス対応マトリックス*』に記載されている NNMi の必要条件に適合
- 以下の追加パッチのインストール
  - *Windows* の場合：
    - *Connecting to SMB share on a Windows 2000-based computer or a Windows Server 2003-based computer may not work with an alias name* に対する Microsoft の修正プログラム。これは以下のサイトからダウンロードできます。
      http://support.microsoft.com/?id=281308
  - *UNIX* の場合：追加要件はありません。
- 特定の HA 製品下で NNMi の実行を予定している場合、その製品のマニュアルに記載されている要件に適合

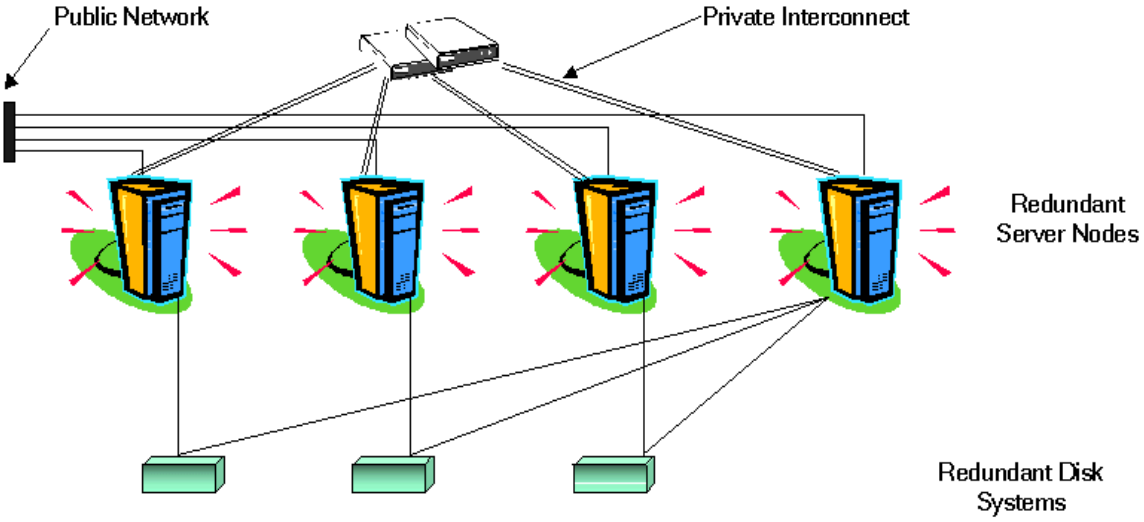
HA 用に NNMi を設定する前に、HA 製品のコマンドを使って、HA クラスタの設定とテストを行います。HA クラスタには、ハートビートチェックやフェイルオーバー起動のような機能が用意されています。HA クラスタ設定には、少なくとも、以下の項目を含める必要があります。

- （UNIX のみ）ssh
- （UNIX のみ）remsh
- DNS で解決可能な、HA クラスタ用の仮想 IP アドレス
- DNS で解決可能な、HA クラスタ用の仮想ホスト名

HA の概念

クラスタアーキテクチャには、クラスタ内の複数のノードのプロセスとリソース用の、単一のグローバルに首尾一貫した管理ビューが備わっています。図 2 に、クラスタアーキテクチャの例を示します。

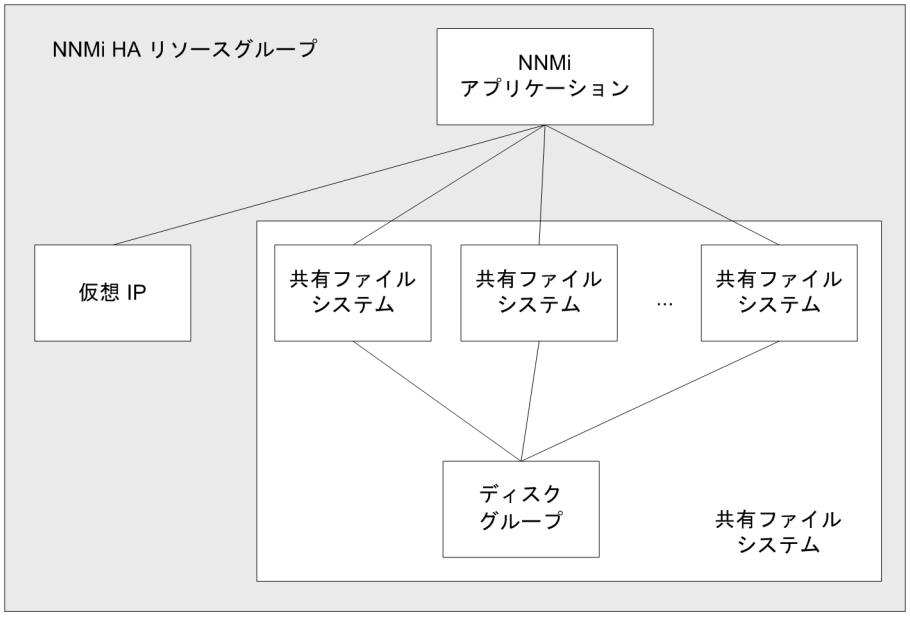
図 2　高可用性クラスタのアーキテクチャ



クラスタ内の各ノードは、1 つ以上の公衆網と 1 つのプライベートインタコネクト（クラスタノード間のデータ伝送用の通信チャネル）に接続されます。

HP ServiceGuard、Veritas Cluster Server、Microsoft Cluster Service などの最新のクラスタ環境では、アプリケーションはリソースの複合体として表現され、単純な操作でアプリケーションをクラスタ環境で実行することができます。リソースは、クラスタ環境で動作するアプリケーションを表す、**HA リソースグループ**に構成されます。図 3 に、HA リソースグループ構成方法の例を示します。

図 3　典型的な HA リソースグループのレイアウト



このマニュアルでは、各種のクラスタ環境内のリソースの集合を指すために、*HA リソースグループ*という用語を使います。各 HA 製品では、HA リソースグループに対して、異なる名前が使われています。表 3 に、このマニュアルの *HA リソースグループ*に相当する、サポート対象の HA 製品で使われている用語をリストします。（各 HA 製品のサポート対象のバージョンについては、サポート対象の HA 製品ページの 78 を参照してください。）

表 3　サポート対象の HA 製品で HA リソースグループに相当する名前

HA 製品	略語	HA リソースグループに相当する名前
HP ServiceGuard	SG	パッケージ
Veritas Cluster Server	VCS	サービスグループ
Microsoft Cluster Services	MSCS	リソースグループ

HA 用語集

表 4 に、一般的な HA 用語の定義を示します。

表 4　一般的な HA 用語

用語	説明
HA リソースグループ	クラスタ環境内で（HA 製品下で）動作するアプリケーションです。HA リソースグループは、同時に、クラスタ内のアプリケーションを表すクラスタオブジェクトでもあります。
ボリュームグループ	1 つの大規模ストレージエリアを形成するよう設定された 1 つ以上のディスクドライブです。
論理ボリューム	ボリュームグループ内で、個別のファイルシステムまたはデバイススワップ空間として使われる任意のサイズの領域です。
プライマリクラスタノード	ソフトウェア製品が最初にインストールされるシステムであり、かつ、HA が最初に設定されるシステムです。 初期セットアップでは、共有ディスクはプライマリクラスタノードにマウントされます。プライマリクラスタノードは、通常、最初のアクティブなクラスタノードになりますが、HA の設定完了後には、プライマリとしての役割を解除できます。HA 設定を変更すると、他のノードをプライマリクラスタノードにできます。
セカンダリクラスタノード	プライマリクラスタノードでの HA 設定の完了後に、HA 設定に追加される任意のシステムです。
アクティブなクラスタノード	現在 HA リソースグループを実行中のシステムです。
パッシブなクラスタノード	HA 用に設定されているが、現在 HA リソースグループを実行していないシステムです。アクティブなクラスタノードで障害が発生すると、HA リソースグループはパッシブなクラスタノードの中で利用可能なノードにフェイルオーバーし、そのノードがアクティブなクラスタノードになります。

NNMi HA クラスタのシナリオ

NNMi HA 設定では、NNMi は各システムにインストールされ、HA リソースグループの一部になります。NNMi データベースは独立したディスクにインストールされ、各システムで動作中の NNMi プログラムからアクセスされます。（任意の時点で共有ディスクにアクセスできるのは、アクティブなクラスタノードである 1 つのシステムだけです。）

このアプローチは、組み込み型のデータベースと他社製データベースソリューションの場合に有効です。

NNMi データベースのバックアップスクリプトとリストアスクリプトを実行できるのは、アクティブなクラスタノードだけです。

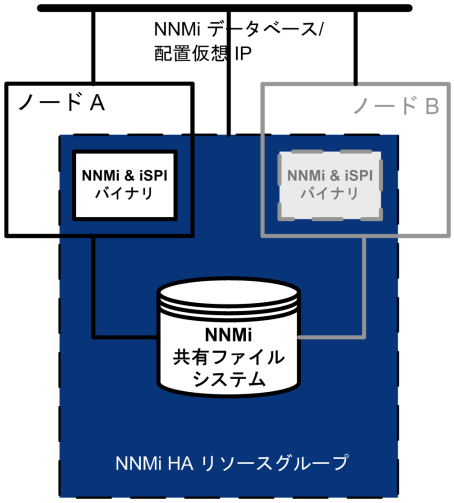
NNMi のみのシナリオ

図 4 に、NNMi HA クラスタのシナリオを図示します。この図では、NNMi HA リソースグループは、NNMi HA クラスタと同義語です。

ノード A とノード B は、いずれもが、すべてのソフトウェアがインストールされた NNMi 管理サーバーであり、そのシステムで実行する NNMi プログラムと iSPI がすべて含まれています。アクティブなクラスタノードが、共有ディスクのランタイムデータにアクセスします。他の製品は、HA リソースグループの仮想 IP アドレスを使って、NNMi に接続します。

クラスタに 3 つ以上の NNMi ノードがある場合は、追加ノードには図 4 のノード B と同様の設定を行います。

図 4　NNMi HA クラスタ用の基本的なシナリオ

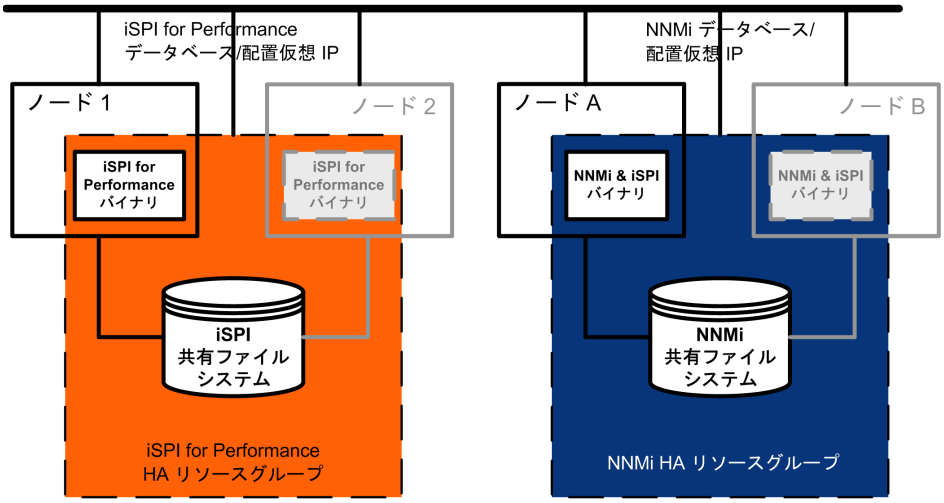


このシナリオの実装方法については、HA 用の NNMi の設定ページの 84 と HA 用の iSPI の設定ページの 89 を参照してください。

専用サーバーのシナリオでの NNMi と NNM iSPI for Performance

NNM iSPI for Performance を専用サーバーで実行する場合は、図 5 のように、この iSPI を NNMi HA クラスタ内で独立した HA リソースグループとして実行するように設定できます。

図 5　専用サーバーでの NNMi と NNM iSPI for Performance 用 HA



このシナリオの実装方法については、HA 用の NNMi の設定ページの 84、HA 用の iSPI の設定ページの 89、および専用サーバーで HA 用に NNM iSPI for Performance を設定するページの 91 を参照してください。

専用サーバーでの NNM iSPI for Performance のその他の選択肢は、以下のとおりです。

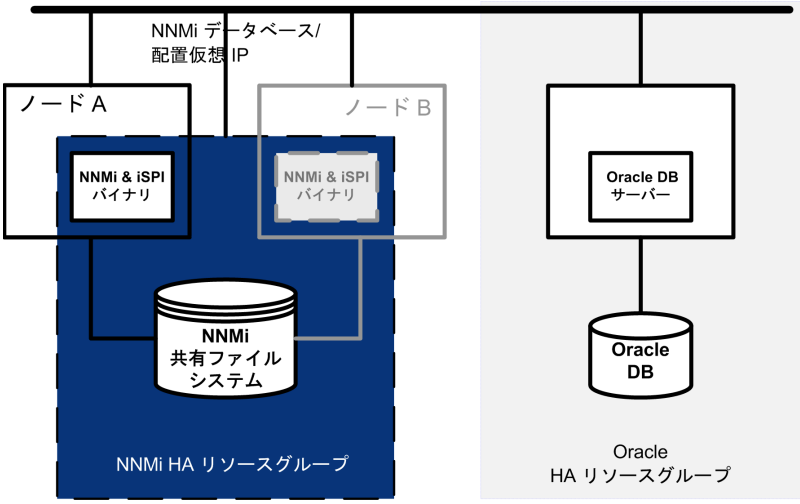
- NNM iSPI for Performance を HA を設定していない単一システムで実行します。このアプローチは、iSPI を評価する場合、あるいは、パフォーマンスデータが必ずしも必要ではない環境で使用します。
- NNM iSPI for Performance を NNMi 用とは異なる HA クラスタ下で実行するように設定します。この場合は、NNM iSPI for Performance の NNMi への依存関係を手動で管理する必要があります。

NNMi で Oracle データベースを使う場合のシナリオ

NNMi 実装で Oracle をメイン NNMi データベースとして使う場合は、Oracle データベースは、パフォーマンス上の理由で、図 6 のように、独立したサーバーにインストールする必要があります。そのため、NNMi HA クラスタでは、次の 2 つの HA リソースグループを設定する必要があります。

- NNMi リソースグループは、NNMi ノードと、Oracle データベースに格納されない NNMi データ用の共有ディスクで構成します。
- Oracle リソースグループは、Oracle データベースサーバーとデータベースディスクで構成します。

図 6　Oracle データベースを使っている NNMi 用の HA

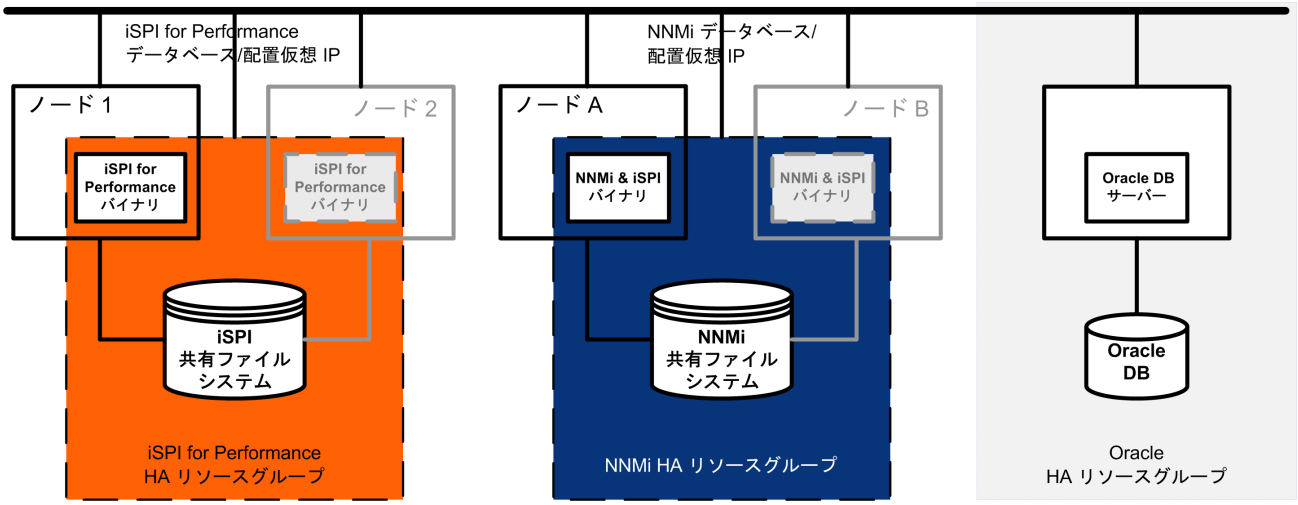


このシナリオの実装方法については、Oracle 環境で、HA 用に NNMi を設定するページの 95 と HA 用の iSPI の設定ページの 89 を参照してください。

NNMi で Oracle データベースを使い、NNM iSPI for Performance を専用サーバーで実行する場合のシナリオ

NNMi 実装で Oracle をメイン NNMi データベースとして使い、NNM iSPI for Performance を専用サーバーで実行する場合は、図 7 のように、NNMi HA クラスタ内に 3 つの HA リソースグループを設定できます。

図 7　NNMi で Oracle データベースを使い、NNM iSPI for Performance を専用サーバーで実行する場合の HA



このシナリオの実装方法については、Oracle 環境で、HA 用に NNMi を設定するページの 95、HA 用の iSPI の設定ページの 89、および専用サーバーで HA 用に NNM iSPI for Performance を設定するページの 91 を参照してください。

マンページ

NNMi マンページには、HA 設定について、以下の内容が含まれています。

- nnm-ha
- nnmhaconfigure.ovpl
- nnmhaunconfigure.ovpl
- nnmhadisk.ovpl
- nnmhaclusterinfo.ovpl
- nnmhastartrg.ovpl
- nnmhastoprg.ovpl

Windows オペレーティングシステムでは、これらのマンページはテキストファイルで提供されます。

HA の設定

HA 用の NNMi の設定

HA 用に NNMi を設定する場合の主な作業は、次の 2 つです。

1 NNMi データファイルを共有ディスクにコピーします。
   - プライマリクラスタノードでの NNMi の設定ページの 86の手順 2～手順 11 に従って、プライマリノードでこの作業を行います。
2 HA 下で NNMi を実行するように、設定します。
   - プライマリクラスタノードでの NNMi の設定ページの 86の手順 12～手順 15 に従って、プライマリノードでこの作業を行います。
   - セカンダリクラスタノードでの NNMi の設定ページの 88 に従って、セカンダリノードでこの作業を行います。

1 つの HA クラスタノードを、プライマリ NNMi 管理サーバーとして割り当てます。これが大部分の時間にアクティブとなるノードです。プライマリノードを設定します。次に HA クラスタ内の残りのすべてのノードをセカンダリノードとして設定します。

HA 用の NNMi の設定は、複数のクラスタノードで同時には*行えません*。1 つのクラスタノードで HA 設定プロセスが完了した後、次のクラスタノードでの HA 設定プロセスを開始するというように、クラスタ環境内のすべてのノードで HA 用に NNMi を設定するまで、この作業を繰り返します。

NNMi HA 設定情報

HA 設定スクリプトは、NNMi HA リソースグループに関する情報を収集します。表 5 に、プライマリノードの設定で必要になる情報を示します。設定作業を開始する前に、これらの情報を用意してください。

表 5　NNMi HA プライマリノードの設定情報

HA 設定項目	説明
HA リソースグループ	NNMi を含む HA クラスタのリソースグループの名前です。 例：nnmtest1
仮想ホストの短い名前	仮想ホストの短い名前です。このホスト名は、HA リソースグループの仮想 IP アドレスにマッピングする必要があります。`nslookup` コマンドで、仮想ホストの短い名前と仮想 IP アドレスを解決できる必要があります。 **注記：**NNMi が仮想ホストの短い名前と仮想 IP アドレスを解決できない場合は、HA 設定スクリプトにより、システムが不安定な状態になる可能性があります。したがって、NNMi HA の設定中に DNS が利用できない場合に備えて、予備のネーミングストラテジ（たとえば、Windows オペレーティングシステムの場合は、`%SystemRoot%\system32\drivers\etc\hosts` ファイルに、UNIX オペレーティングシステムの場合は、`/etc/hosts` ファイルに、それぞれ情報を記述する）を用意しておくことをお勧めします。
仮想ホストのネットマスク	仮想ホスト IP アドレスで使われるサブネットマスクです。

表 5　NNMi HA プライマリノードの設定情報

HA 設定項目	説明
仮想ホストのネットワークインタフェース	仮想ホスト IP アドレスが使われるネットワークインタフェースです。例： • *Windows* の場合：ローカル エリア接続 • *HP-UX* の場合：lan0 • *Linux* の場合：eth0 • *Solaris* の場合：bge0
共有ファイルシステムのタイプ	HA リソースグループで使われる共有ディスクの設定タイプです。取り得る値は以下のとおりです。 • `disk`：共有ディスクは、標準のファイルシステムタイプを使う、物理的に接続されたディスクです。HA 設定スクリプトは、共有ディスクを設定できます。詳細については、この表のファイルシステムタイプの欄を参照してください。 • `none`：共有ディスクには、`disk` オプションで説明している設定以外の SAN や NFS 構成などを使います。HA 設定スクリプトを実行すると、共有ディスクの設定ページの 97 のように、共有ディスクが設定されます。
ファイルシステムタイプ	共有ディスクのファイルシステムタイプです（共有ファイルシステムのタイプが `disk` の場合）。HA 設定スクリプトは、ディスクの検証方法を調べるために、この値を HA 製品に渡します。 以下の共有ディスクフォーマットはテスト済みです。 • *Windows* の場合：基本型 (Windows 2003 での共有ディスク設定についての注記ページの 98 を参照) • *HP-UX* の場合：vxfs • *Linux* の場合：lvm2 • *Solaris* の場合：vxfs 他のフォーマットの物理的に接続されたディスクも、この HA 実装で動作すると考えています。そのため、標準のファイルシステムタイプを使っている、物理的に接続されたディスクを持つ NNMi HA システムの分析についても、限定的にサポートしています。 **注記**：テストされていない共有ディスクフォーマットを使っている場合は、NNMi HA 設定スクリプトによって挿入された値が正しいことを確認するために、リソースグループ設定を調べてください。
ディスクグループ	(UNIX のみ) NNMi 共有ファイルシステムのディスクグループの名前です。この名前は、HA リソースグループの名前をベースにします。 例：nnmtest1-dg
ボリュームグループ	(UNIX のみ) NNMi 共有ファイルシステムのボリュームグループの名前です。この名前は、HA リソースグループの名前をベースにします。 例：nnmtest1-vol
マウントポイント	NNMi の共有ディスクをマウントするディレクトリの場所です。このマウントポイントは、すべてのシステムで同じである必要があります。(つまり、各ノードでは、マウントポイントに同じ名前を使う必要があります。) 例： • *Windows* の場合：`S:\` • *UNIX* の場合：`/nnmmount`

プライマリクラスタノードでの NNMi の設定

プライマリクラスタノードで以下の手順を実行します。

1 システムが、HA 用に NNMi を設定するための前提条件ページの 78 に示されているすべての要件を満たしていることを確認します。

2 NNMi がインストールされていない場合は、インストールします。そして、NNMi が正しく動作することを確認します。

この時点では iSPI のインストールは不要ですが、NNMi 管理サーバーにインストール済みの iSPI は残しておいて構いません。

3 次のコマンドを使って、NNMi 設定をバックアップします。

```
nnmbackup.ovpl -scope config -target <directory>
```

このコマンドの詳細については、NNMi のバックアップおよびリストアツールページの 123 を参照してください。

4 以下のファイルを別の場所にコピーして、NNMi のライセンスファイルをバックアップします。

- *Windows* の場合：`%AUTOPASS_HOME%\data\LicFile.txt`

  `%AUTOPASS_HOME%` の値を調べるには、システム環境変数を参照します。

- *UNIX* の場合：`/var/opt/OV/HPOvLIC/LicFile.txt`

5 NNMi HA リソースグループ用に、少なくとも 1 つの共有ディスクを含む、ディスクデバイスグループ（および論理ボリューム）を定義します。例：

- *Windows* の場合：ディスクの管理を使って、ディスクのマウントポイントを設定し、ディスクをフォーマットします。
- *HP-UX* と *Linux* の場合：

```
pvcreate -f /dev/dsk/<physical_disk_name>
mkdir /dev/<disk_group>
mknod c 64 0x00h0 /dev/<disk_group>/group
vgcreate /dev/rdsk/<physical_disk_name> <disk_group_name>
lvcreate -L <disk_size> -n <logical_volume> <disk_group>
newfs /dev/<disk_group>/r<logical_volume>
```

- *Solaris* の場合：次のように、Symantec Veritas Storage Foundation を使います。

  `vxdiskadm` を使って、ディスクを追加し、初期化します。

  `vxassist make` を使って、領域ごとにディスクを割り当てます。

```
mkfs -F vxfs /dev/vx/dsk/<disk_group>/<logical_volume_group>
```

UNIX オペレーティング システムの参考 Web サイトは、次のとおりです。
http://www.unixguide.net/unixguide.shtml

6 共有ディスクのマウント ポイントになるディレクトリを作成しま（たとえば、`S:\` または `/nnmmount`）。

- *Windows* の場合：Windows Explorer とディスクの管理を使います。
- *UNIX* の場合：共有ディスクのマウントポイントディレクトリが、ユーザーは `root`、グループは `sys` で作成され、パーミッションには `555` が設定されていることを確認します。例：

```
ls -l /nnmmount
```

7 共有ディスクをマウントします。例：

- *Windows* の場合：ディスクの管理を使います。
- *HP-UX* の場合：

  `mount /dev/<disk_group>/<logical_volume> /nnmmount`

- *Linux* の場合：

  `mount /dev/<disk_group>/<logical_volume_group> /nnmmount`

- *Solaris* の場合：

  `vxfs: mount /dev/vx/dsk/<disk_group>/<volume_group> /nnmmount`

8 NNMi を停止します。

```
ovstop -c
```

9 NNMi データベースを共有ディスクにコピーします。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM
-to <HA_mount_point>
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM
-to <HA_mount_point>
```

データベースの破壊を避けるために、この（-to オプションを指定した）コマンドは一回しか実行できません。代替方法については、すべてのクラスタノードの設定解除後に、HA 用の NNMi を再び有効にするページの 115 を参照してください。

10 共有ディスクをマウント解除します。

11 （UNIX のみ）ディスクグループを非アクティブ化します。

```
umount <HA_mount_point>
```

12 NNMi が実行中でないことを確認します。

```
ovstop -c
```

13 NNMi HA リソースグループを設定します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl NNM
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl NNM
```

84 ページの表 5 に、このコマンドで必要な情報を示します。

14 手順 13 で共有ファイルシステムタイプとして指定した値に応じて、手順が異なります（84 ページの表 5 の共有ファイルシステムのタイプとファイルシステムタイプ）。

- タイプ disk を指定した場合は、nnmhaconfigure.ovpl コマンドによって、共有ディスクが設定されています。手順 15 に進みます。
- タイプ none を指定した場合は、共有ディスクの設定ページの 97 に従って共有ディスクを設定し、手順 15 に進みます。

15 NNMi HA リソースグループを起動します。

- *Windows* の場合：

  %NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM *<resource_group>*

- *UNIX* の場合：

  $NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM *<resource_group>*

NNMi を正常に起動できなかった場合は、HA 設定のトラブルシューティングページの 113 を参照してください。

これで、NNMi が HA 下で動作するようになりました。通常のオペレーションでは、`ovstart` コマンドや `ovstop` コマンドは*使わない*でください。これらのコマンドを使うのは、HA のメンテナンスが目的で、使うことが指示された場合だけです。

セカンダリクラスタノードでの NNMi の設定

セカンダリクラスタノードでは 1 つのノードごとに順番に以下の手順を実行します。

1 プライマリクラスタノードでの NNMi の設定ページの 86の作業を完了していない場合は、完了させます。

2 システムが、HA 用に NNMi を設定するための前提条件ページの 78 に示されているすべての要件を満たしていることを確認します。

3 NNMi がインストールされていない場合は、NNMi をインストールし、正しく動作することを確認します。

   この時点では iSPI のインストールは不要ですが、NNMi 管理サーバーにインストール済みの iSPI は残しておいて構いません。

4 共有ディスクのマウントポイントを作成します（たとえば、`S:\` または `/nnmmount`）。

このマウントポイントでは、手順プライマリクラスタノードでの NNMi の設定の手順 6で作成したマウントポイントと同じ名前を使う必要があります。

5 NNMi HA リソースグループを設定します。

- *Windows* の場合：
  %NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl NNM
- *UNIX* の場合：
  $NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl NNM

コマンドの要求に応じて、HA リソースグループ名を指定します。

6 設定が正常に行われたことを確認します。

- *Windows* の場合：

  %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl -group *<resource_group>* -nodes

- *UNIX* の場合：

  $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -group *<resource_group>* -nodes

このコマンドの出力には、指定した HA リソースグループに設定されたすべてのノードがリストされます。

7 オプションで、プライマリノードのリソースグループをオフラインにし、セカンダリノードのリソースグループをオンラインにすることで、設定をテストします。

HA 用の iSPI の設定

iSPI HA 設定情報

NNMi バージョン 8.10 から、NNMi に付属した HA 設定コマンドを使って、iSPI を NNMi HA リソースグループ内で実行するように設定できます。各 iSPI は、以下の要件を満たす必要があります。

- iSPI は NNMi 管理サーバー上で実行する。
- iSPI は NNMi と同じ Postgres インスタンスを使用する。

NNM iSPI for Performance を別のシステムで実行する場合は、専用サーバーで HA 用に NNM iSPI for Performance を設定するページの 91 を参照してください。

この設定手順では、iSPI が NNMi 管理サーバーにインストールする Perl モジュールのベース名が必要になります。表 6 に、この章を最後にアップデートしたときにわかっていた名前をリストします。他の iSPI で使う値を調べるには、統合とプラグイン章の該当する iSPI の項を参照してください。

表 6　iSPI の Perl モジュール名

iSPI	Perl モジュールのベース名
NNM iSPI for Performance	`PerfSPIHA`

HA 用の iSPI の設定順序に依存関係はありません。NNMi 管理サーバーで複数の iSPI を実行することを予定している場合は、以下のプロセスフローのいずれかを実行します。

- 最初の iSPI については、アドオン iSPI の設定（他のすべての iSPI）の手順 3 と手順 4 の両方を実行します。次に 2 番目以降の iSPI についても、手順 3 と手順 4 の両方を実行します。
- すべての iSPI について、アドオン iSPI の設定（他のすべての iSPI）の手順 3 を実行し、パッシブなクラスタノードで 1 つのノードごとに順番に手順 4 を実行します。

NNM iSPI for Performance をアドオン iSPI として設定

同一システムの NNM iSPI for Performance を NNMi HA リソースグループの下で実行するための最も簡単な手順は、以下のとおりです。

1 NNMi HA クラスタ内のすべてのノードで、HA 用の NNMi の設定ページの 84 または Oracle 環境で、HA 用に NNMi を設定するページの 95 に従って、NNMi HA リソースグループを完全に設定し、起動します。クラスタが、88 ページの手順 6 のように、正しく設定されたことを確認します。

iSPI を HA 下で実行するように設定するためのツールは、設定作業中に NNMi と対話します。設定作業を簡単にするために、すべての iSPI の設定作業中は、NNMi HA クラスタ内のすべてのノードを稼働中にしておいてください。

2 アクティブなノードで、NNMi サービスが実行中であることを確認します。

```
ovstatus -c
```

すべての NNMi サービスで、［実行中］状態が表示される必要があります。

3 NNMi HA クラスタ内のアクティブなノードで、以下の手順を実行します。

a NNM iSPI for Performance イネーブルメントスクリプトを実行します。

b NNM iSPI for Performance をインストールします。

以下の手順についての詳細は、『*NNM iSPI for Performance インストールガイド*』を参照してください。

4 NNMi HA クラスタ内のパッシブな各ノードで、以下の手順を実行します。

a NNM iSPI for Performance イネーブルメントスクリプトを実行します。

b NNM iSPI for Performance をインストールします。

アドオン iSPI の設定（他のすべての iSPI）

ベストプラクティス

NNM iSPI for Performance のインストール作業では、アドオン iSPI が NNMi HA リソースグループ下で実行されるように、自動的に設定されます。NNM iSPI for Performance をインストールしていない場合は、NNM iSPI for Performance をアドオン iSPI として設定を参照してください。

アドオン iSPI が HA リソースグループ下で実行されるように設定するには、以下の手順を実行します。

1 NNMi HA クラスタ内のすべてのノードで、HA 用の NNMi の設定ページの 84 または Oracle 環境で、HA 用に NNMi を設定するページの 95 に従って、NNMi HA リソースグループを完全に設定し、起動します。クラスタが、88 ページの手順 6 のように、正しく設定されたことを確認します。

iSPI を HA 下で実行するように設定するためのツールは、設定作業中に NNMi と対話します。設定作業を簡単にするために、すべての iSPI の設定作業中は、NNMi HA クラスタ内のすべてのノードを稼働中にしておいてください。

2 NNMi HA クラスタ内のアクティブなノードで、NNMi サービスが実行中であることを確認します。

```
ovstatus -c
```

すべての NNMi サービスで、［実行中］状態が表示される必要があります。

3 NNMi HA クラスタ内のアクティブなノードで、HA 用の iSPI を設定します。

a iSPI をインストールしていない場合は、インストールします。

b iSPI を NNMi HA リソースグループに追加します。

— *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl NNM
-addon <iSPI_PM_Name>
```

— *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl NNM
-addon <iSPI_PM_Name>
```

<iSPI_PM_Name> には、iSPI が NNMi 管理サーバーにインストールする Perl モジュールのベース名を指定します。*<iSPI_PM_Name>* に指定する値を調べるには、89 ページの表 6、または統合とプラグイン章の該当する iSPI の項を参照してください。

4 NNMi HA クラスタ内のパッシブな各ノードで、HA 用の iSPI を設定します。

a iSPI をインストールしていない場合は、インストールします。

b iSPI を NNMi HA リソースグループに追加します。

— *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl NNM
-addon <iSPI_PM_Name>
```

— *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl NNM
-addon <iSPI_PM_Name>
```

5 設定を確認します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl
-config NNM -get NNM_ADD_ON_PRODUCTS
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl
-config NNM -get NNM_ADD_ON_PRODUCTS
```

コマンドの出力には、アドオン iSPI の設定が *<iSPI_PM_Name>*[*hostname_list*] のフォーマットでリストされます。例：

```
PerfSPIHA [hostname1, hostname2]
```

専用サーバーで HA 用に NNM iSPI for Performance を設定する

NNM iSPI for Performance HA 設定情報

HA 設定スクリプトは、NNM iSPI for Performance HA リソースグループに関する情報を収集します。表 7 に、プライマリノードの設定で必要になる情報を示します。設定作業を開始する前に、これらの情報を用意してください。

表 7　NNM iSPI for Performance HA プライマリノードの設定情報

HA 設定項目	説明
HA リソースグループ	NNM iSPI for Performance を含む HA クラスタのリソースグループの名前です。 例：perftest1
仮想ホストの短い名前	仮想ホストの短い名前です。このホスト名は、HA リソースグループの仮想 IP アドレスにマッピングする必要があります。`nslookup` コマンドで、仮想ホストの短い名前と仮想 IP アドレスを解決できる必要があります。 **注記：**NNMi が仮想ホストの短い名前と仮想 IP アドレスを解決できない場合は、HA 設定スクリプトにより、システムが不安定な状態になる可能性があります。したがって、NNM iSPI for Performance HA の設定中に DNS が利用できない場合に備えて、予備のネーミングストラテジ（たとえば、Windows オペレーティングシステムの場合は、`%SystemRoot%\system32\drivers\etc\hosts` ファイルに、UNIX オペレーティングシステムの場合は、`/etc/hosts` ファイルに、それぞれ情報を記述する）を用意しておくことをお勧めします。
仮想ホストのネットマスク	仮想ホスト IP アドレスで使われるサブネットマスクです。
仮想ホストのネットワークインタフェース	仮想ホスト IP アドレスが使われるネットワークインタフェースです。例： • *Windows* の場合：ローカルエリア接続 • *HP-UX* の場合：lan0 • *Linux* の場合：eth0 • *Solaris* の場合：bge0
共有ファイルシステムのタイプ	HA リソースグループで使われる共有ディスクの設定タイプです。取り得る値は以下のとおりです。 • `disk`：共有ディスクは、標準のファイルシステムタイプを使う、物理的に接続されたディスクです。HA 設定スクリプトは、共有ディスクを設定できます。詳細については、この表のファイルシステムタイプの欄を参照してください。 • `none`：共有ディスクには、`disk` オプションで説明している設定以外の SAN や NFS 構成などを使います。HA 設定スクリプトを実行すると、共有ディスクの設定ページの 97 のように、共有ディスクが設定されます。

表 7　NNM iSPI for Performance HA プライマリノードの設定情報

HA 設定項目	説明
ファイルシステムタイプ	共有ディスクのファイルシステムタイプです（共有ファイルシステムのタイプが disk の場合）。HA 設定スクリプトは、ディスクの検証方法を調べるために、この値を HA 製品に渡します。 以下の共有ディスクフォーマットはテスト済みです。 • *Windows* の場合：基本型 (Windows 2003 での共有ディスク設定についての注記ページの 98 を参照) • *HP-UX* の場合：vxfs • *Linux* の場合：lvm2 • *Solaris* の場合：vxfs 他のフォーマットの物理的に接続されたディスクも、この HA 実装で動作すると考えています。そのため、標準のファイルシステムタイプを使っている、物理的に接続されたディスクを持つ NNMi HA システムの分析についても、限定的にサポートしています。 **注記：**テストされていない共有ディスクフォーマットを使っている場合は、NNMi HA 設定スクリプトによって挿入された値が正しいことを確認するために、リソースグループ設定を調べてください。
ディスクグループ	（UNIX のみ）NNM iSPI for Performance 共有ファイルシステムのディスクグループの名前です。この名前は、HA リソースグループの名前をベースにします。 例：perftest1-dg
ボリュームグループ	（UNIX のみ）NNM iSPI for Performance 共有ファイルシステムのボリュームグループの名前です。この名前は、HA リソースグループの名前をベースにします。 例：perftest1-vol
マウントポイント	NNM iSPI for Performance の共有ディスクをマウントするディレクトリの場所です。このマウントポイントは、すべてのシステムで同じである必要があります。（つまり、各ノードでは、マウントポイントに同じ名前を使う必要があります。）例： • *Windows* の場合：`P:\` • *UNIX* の場合：`/perfmount`

プライマリクラスタノードでの NNM iSPI for Performance の設定

ベストプラクティス

NNMi HA クラスタ内の各ノードで NNM iSPI for Performance イネーブルメントスクリプトを実行していなかった場合は、実行します。

プライマリクラスタノードで以下の手順を実行します。

1. NNM iSPI for Performance をインストールしていなかった場合は、インストールします。［**NNM iSPI for Performance の設定**］ウィンドウに必要な情報を入力します。ただし、サービスは*起動しません*。

   このウィンドウ内のフィールドについての詳細は、『*NNM iSPI for Performance インストールガイド*』を参照してください。

2. パフォーマンス HA リソースグループ用に、少なくとも 1 つの共有ディスクを含む、ディスクデバイスグループ（および論理ボリューム）を定義します。詳細については、86 ページの手順 5 を参照してください。

3. 共有ディスクのマウントポイントになるディレクトリを作成します（たとえば、`P:\` または `/perfmount`）。たとえば、86 ページの手順 6 を参照してください。

4 共有ディスクをマウントします。たとえば、87 ページの手順 7 を参照してください。

5 NNM iSPI for Performance が実行中でないことを確認します。

- *Windows* の場合：

```
%NnmInstallDir%\NNMPerformanceSPI\bin\statusSPI.ovpl
%NnmInstallDir%\NNMPerformanceSPI\bin\statusERS.ovpl
```

- *UNIX* の場合：

```
$NnmInstallDir/NNMPerformanceSPI/bin/statusSPI.ovpl
$NnmInstallDir/NNMPerformanceSPI/bin/statusERS.ovpl
```

6 パフォーマンス HA リソースグループを設定します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl PerfSPIHA
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl PerfSPIHA
```

92 ページの表 7 に、このコマンドで必要な情報を示します。

7 手順 6 で共有ファイルシステムタイプとして指定した値に応じて、手順が異なります（92 ページの表 7 の共有ファイルシステムのタイプとファイルシステムタイプ）。

- タイプ `disk` を指定した場合は、`nnmhaconfigure.ovpl` コマンドによって、共有ディスクが設定されています。手順 8 に進みます。
- タイプ `none` を指定した場合は、共有ディスクの設定ページの 97 に従って共有ディスクを設定し、手順 8 に進みます。

8 パフォーマンス HA リソースグループを起動します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl PerfSPIHA
<resource_group>
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl PerfSPIHA
<resource_group>
```

NNM iSPI for Performance を正常に起動できなかった場合は、一般的な HA のトラブルシューティングページの 113 を参照してください。

セカンダリクラスタノードに NNM iSPI for Performance を設定する

セカンダリクラスタノードでは 1 つのノードごとに順番で以下の手順を実行します。

1 プライマリクラスタノードでの NNM iSPI for Performance の設定ページの 93 の作業を完了していない場合は、完了させます。

2 共有ディスクのマウントポイントを作成します（たとえば、`P:\` または `/perfmount`）。

このマウントポイントでは、手順プライマリクラスタノードでの NNM iSPI for Performance の設定の手順 2 で作成したマウントポイントと同じ名前を使う必要があります。

3 NNM iSPI for Performance HA リソースグループを設定します。

- *Windows* の場合：

  %NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl PerfSPIHA

- *UNIX* の場合：

  $NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl PerfSPIHA

コマンドの要求に応じて、パフォーマンス HA リソースグループ名を指定します。

4 設定が正常に行われたことを確認します。

- *Windows* の場合：

  %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl -group <*resource_group*> -nodes

- *UNIX* の場合：

  $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -group <*resource_group*> -nodes

このコマンドの出力には、指定した HA リソースグループに設定されたすべてのノードがリストされます。

5 オプションで、プライマリノードのリソースグループをオフラインにし、セカンダリノードのリソースグループをオンラインにすることで、設定をテストします。

Oracle 環境で、HA 用に NNMi を設定する

ここでは、Oracle データベースを使っている NNMi を HA 下で実行するための設定作業の概要を説明します。Oracle の設定方法は多数あり、Oracle のリリースによっても異なります。Oracle を HA 下で実行するための設定方法と Oracle HA リソースグループでの NNMi の依存関係の作成方法については、HA 製品のマニュアルの Oracle データベースのサポートに関する説明を参照してください。Oracle の Web サイト (**http://www.oracle.com**) でも、HA 製品用の Oracle 設定方法が紹介されています。

Oracle を使っている NNMi の HA 設定情報

Oracle と NNMi の両方を HA 下で実行する場合は、それぞれの製品を異なる HA リソースグループに設定する必要があります。Oracle リソースグループは、NNMi リソースグループを起動する前に、完全な起動状態になっている必要があります。両方のリソースグループが同じ HA クラスタに含まれている場合は、クラスタ設定を変更してリソースグループの起動順序を設定します。それぞれのリソースグループが異なる HA クラスタに含まれている場合は、Oracle リソースグループでの NNMi リソースグループの依存関係を確認します。

NNMi リソースグループには、Oracle データベースに格納されていない NNMi データ用の共有ディスクを含む必要があります。

Oracle を使っている NNMi に HA を設定する

1 Oracle を HA 下で実行することを予定している場合は、最初に、以下の手順を実行します。

2 プライマリ NNMi ノードに、NNMi をインストールします。インストール時には、Oracle HA リソースグループ用の仮想 IP アドレスまたは仮想ホスト名を使用します。

3 プライマリ NNMi ノードで、プライマリクラスタノードでの NNMi の設定ページの 86 に従って、NNMi を HA 下で実行できるように設定します。

4 Oracle HA リソースグループでの NNMi の依存関係を設定します。

具体的な手順については、HA 製品のマニュアルを参照してください。

5 セカンダリ NNMi ノードに、NNMi をインストールします。インストール時には、Oracle HA リソースグループ用の仮想 IP アドレスまたは仮想ホスト名を使用します。

6 セカンダリ NNMi ノードで、セカンダリクラスタノードでの NNMi の設定ページの 88 に従って、NNMi を HA 下で実行するように設定します。

7 各セカンダリ NNMi ノードで、手順 5 と手順 6 を繰り返します。

共有 NNMi データ

HA 下で実行する NNMi 実装では、HA クラスタ内のすべての NNMi ノード間でファイルを共有するために、独立したディスクを使う必要があります。

➤ Oracle をプライマリデータベースとして使っている NNMi の実装でも、共有データ用に独立したディスクを使う必要があります。

NNMi の共有ディスク内のデータ

ここでは、NNMi を HA 下で実行する場合に、共有ディスクで管理される NNMi のデータファイルをリストします。

ファイルの場所は、次のように、共有ディスク内の場所にマッピングされます。

- *Windows* の場合：
  - `%NnmInstallDir%` は、`%HA_MOUNT_POINT%\NNM\installDir` にマッピングされます。
  - `%NnmDataDir%` は、`%HA_MOUNT_POINT%\NNM\dataDir` にマッピングされます。
- *UNIX* の場合：
  - `$NnmInstallDir` は、`$HA_MOUNT_POINT/NNM/installDir` にマッピングされます。
  - `$NnmDataDir` は、`$HA_MOUNT_POINT/NNM/dataDir` にマッピングされます。

共有ディスクに移動されるディレクトリは、以下のとおりです。

- *Windows* の場合：
  - `%NnmDataDir%\shared\nnm\databases\Postgres`
    組み込みデータベース。すべての実装に存在します。
  - `%NnmDataDir%\log\nnm`
    NNMi のロギングディレクトリ。
  - `%NnmDataDir%\shared\nnm\databases\eventdb`
    pmd イベントデータベース。
  - `%NnmInstallDir%\nonOV\jboss\nms\server\nms\data`
    ovjboss で使われるトランザクションストア。

- *UNIX* の場合 :
  - `$NnmDataDir/shared/nnm/databases/Postgres`
    組み込みデータベース。すべての実装に存在します。
  - `$NnmDataDir/log/nnm`
    NNMi のロギングディレクトリ。
  - `$NnmDataDir/shared/nnm/databases/eventdb`
    pmd イベントデータベース。
  - `$NnmInstallDir/nonOV/jboss/nms/server/nms/data`
    ovjboss で使われるトランザクションストア。

これらのファイルは、`nnmhadisk.ovpl` コマンドによって、共有ディスク間でコピーされます。この章の手順に従って、このコマンドを実行します。コマンド構文の概要については、*nnm-ha* のマンページを参照してください。

ファイルレプリケーションの設定

NNMi HA の実装では、ファイルレプリケーションを使って、HA クラスタ内のすべての NNMi ノードの NNMi 設定ファイルのコピーを管理します。デフォルトでは、NNMi コマンドの `nnmdatareplicator.ovpl` で、ファイルレプリケーションを管理します。このコマンドは、フェイルオーバープロセス中に、アクティブノードからパッシブノードに NNMi 設定ファイルをコピーします。`nnmdatareplicator.conf` ファイルには、データレプリケーションに含める NNMi のフォルダーとファイルを指定します。

データレプリケーションプロセスの詳細については、*nnm-ha* のマンページを参照してください。

共有ディスクの設定

共有ディスクがテスト済みのフォーマット（84 ページの表 5 にリスト）の場合は、HA 設定スクリプトによって共有ディスクが準備されるので、以下の手順はスキップします。

共有ディスクが、未検証の SAN や NFS 構成などを使っている場合は、共有ディスクを手動で準備する必要があります。HA の設定中にファイルシステムタイプに値 `none` を指定し、製品マニュアルに従ってディスク設定ファイルを変更します。例 :

- *HP-UX* の場合 :

  `/etc/cmcluster/<resource_group>/<resource_group>.cntl`

- *Linux* の場合 :

  `/usr/local/cmcluster/conf/<resource_group>/<resource_group>.cntl`

- *Solaris* の場合 : ディスク エントリーを追加し、`/opt/VRTSvcs/bin/hares` コマンドを使って、HA 設定ファイルにリンクします。
- *Windows* の場合 : クラスタアドミニストレータ (`cluadmin.exe`)、または `cluster.exe` コマンドを使って、リソースグループにリソースを追加します。

Windows 2003 での共有ディスク設定についての注記

Microsoft Knowledge Base の文書 237853 によれば、Microsoft Cluster Services を使っている Windows 2003 では、ダイナミックディスクはサポートされていません。正しくディスクを設定するには、以下の Web サイトの情報を参照してください。

- **http://support.microsoft.com/kb/237853**
- **http://www.petri.co.il/difference_between_basic_and_dynamic_disks_in_windows_xp_2000_2003.htm**

HA クラスタ内の NNMi のライセンス契約

NNMi を HA 下で実行する場合は、NNMi のライセンスはクラスタの仮想 IP アドレスが対象になります。したがって、HA クラスタでは 1 つの NNMi ライセンスしか必要ではありません。

HA 用に NNMi ライセンスをインストールするには、アクティブな NNMi クラスタノードで、以下の手順を実行します。

1. コマンドプロンプトで、以下のコマンドを入力します。
   - *Windows* の場合：**%NnmInstallDir%\bin\nnmlicense.ovpl NNM -g**
   - *UNIX* の場合：**$NnmInstallDir/bin/nnmlicense.ovpl NNM -g**
2. ［**License Password**］ダイアログボックスで、［**Request License**］をクリックします。
3. 画面の指示に従って、HA クラスタの仮想 IP アドレスに対応する、恒久ライセンスのパスワードを取得します。
4. コマンドプロンプトで以下のコマンドを入力し、システムをアップデートして、ライセンスデータファイルを格納します。
   - *Windows* の場合：

     %NnmInstallDir%\bin\nnmlicense.ovpl NNM -f *<license_file>*
   - *UNIX* の場合：

     $NnmInstallDir/bin/nnmlicense.ovpl NNM -f *<license_file>*
5. 共有ディスクの NNM ディレクトリ内のファイル licenses.txt（たとえば、S:\NNM\licenses.txt や /nnmount/NNM/licenses.txt）を、アクティブノードの次の場所にあるライセンスファイル内の新しい情報を使ってアップデートします。
   - *Windows* の場合：%AUTOPASS_HOME%\data\LicFile.txt

     %AUTOPASS_HOME% の値を調べるには、システム環境変数を参照します。
   - *UNIX* の場合：/var/opt/OV/HPOvLIC/LicFile.txt

   次のいずれかを行います。
   - このファイルが共有ディスクにある場合は、アクティブノードの LicFile.txt 内の新しいライセンスキーを共有ディスクの licenses.txt に追加します。
   - このファイルが共有ディスクにない場合は、LicFile.txt をアクティブノードから共有ディスクの NNM ディレクトリ内の licenses.txt にコピーします。

HA 設定のメンテナンス

メンテナンスモード

HA リソースグループをメンテナンスモードにする

HA リソースグループをメンテナンスモードにすると、HA リソースグループのモニタリングが無効になります。HA リソースグループがメンテナンスモードになっていると、その HA リソースグループの製品の停止や起動を行ってもフェイルオーバーは行われません。

HA リソースグループをメンテナンスモードにするには、アクティブノードで以下のファイルを作成します。

- *Windows* の場合：
  `%NnmDataDir%\hacluster\<resource_group>\maintenance`
- *UNIX* の場合：`$NnmDataDir/hacluster/<resource_group>/maintenance`

  ファイルは空で構いません。

NNMi 8.0x の場合は、メンテナンスファイルは以下のとおりです。

- *Windows* の場合：`%NnmDataDir%\hacluster\maint_NNM`
- *UNIX* の場合：`$NnmDataDir/hacluster/<resource_group>/maint_NNM`

HA リソースグループのメンテナンスモードを解除する

HA リソースグループのメンテナンスモードを解除すると、HA リソースグループのモニタリングが再び有効になります。HA リソースグループの製品を停止すると、HA リソースグループはパッシブなクラスタノードへフェイルオーバーします。

HA リソースグループのメンテナンスモードを解除するには、メンテナンスを開始するまでアクティブだったクラスタノードからメンテナンスファイルを削除します。このファイルについては、HA リソースグループをメンテナンスモードにする を参照してください。

HA クラスタ内の NNMi のメンテナンス

NNMi の起動と停止

NNMi を HA 下で実行している場合は、HA のメンテナンスが目的の指示がない限り、`ovstart` コマンドや `ovstop` コマンドは、*使わない*でください。通常のオペレーションでは、NNMi に用意されている HA コマンドまたは HA 製品の適切なコマンドを使って、リソースグループの起動や停止を行います。

クラスタ環境で NNMi のホスト名や IP アドレスを変更する

クラスタ環境内のノードは、複数の IP アドレスやホスト名を持つことができます。ノードが別のサブネットのメンバーになった場合は、IP アドレスを変更する必要があります。それにより、IP アドレスや完全修飾ドメイン名が変更されます。

たとえば、UNIX システムでは、IP アドレスと関連ホスト名は、通常、次のいずれかの方法を使って設定されています。

- /etc/hosts
- ドメインネームサービス (DNS)
- ネットワーク情報サービス（HP-UX または Linux では NIS、Solaris では NIS+）

NNMi は、管理対象ノードが参照できるように、NNMi データベース内に管理サーバーのホスト名と IP アドレスを格納します。

ネームサーバーがない環境からネームサーバー（すなわち、DNS や BIND）がある環境に移行した場合は、ネームサーバーが新しい IP アドレスを解決することを確認してください。

ホスト名は、IP ネットワーク内で管理対象ノードを特定するために使われます。ノードには複数の IP アドレスが設定できますが、ホスト名は特定のノードを指定するために使われます。システムのホスト名は、`hostname` コマンドを使ったときに返される文字列です。

管理サーバーのホスト名または IP アドレスを変更するには、アクティブな NNMi クラスタノードで以下の手順を実行します。

1. コマンドプロンプトで、以下のコマンドを入力します。
   - *Windows* の場合：**%NnmInstallDir%\bin\nnmlicense.ovpl NNM -g**
   - *UNIX* の場合：**$NnmInstallDir/bin/nnmlicense.ovpl NNM -g**
2. ［**License Password**］ダイアログボックスで、［**Request License**］をクリックします。
3. 画面の指示に従って、HA クラスタの新しい仮想 IP アドレスに対応する、恒久ライセンスのパスワードを取得します。
4. コマンドプロンプトで以下のコマンドを入力し、システムをアップデートして、ライセンスデータファイルを格納します。
   - *Windows* の場合：

     %NnmInstallDir%\bin\nnmlicense.ovpl NNM -f *<license_file>*
   - *UNIX* の場合：

     $NnmInstallDir/bin/nnmlicense.ovpl NNM -f *<license_file>*
5. 共有ディスクの NNM ディレクトリ内のファイル `licenses.txt`（たとえば、`S:\NNM\licenses.txt` や `/nnmount/NNM/licenses.txt`）を、アクティブノードの次の場所にあるライセンスファイル内の新しい情報を使ってアップデートします。
   - *Windows* の場合：`%AUTOPASS_HOME%\data\LicFile.txt`

     `%AUTOPASS_HOME%` の値を調べるには、システム環境変数を参照します。
   - *UNIX* の場合：`/var/opt/OV/HPOvLIC/LicFile.txt`

   次のいずれかを行います。
   - このファイルが共有ディスクにある場合は、アクティブノードの `LicFile.txt` 内の新しいライセンスキーを共有ディスクの `licenses.txt` に追加します。
   - このファイルが共有ディスクにない場合は、`LicFile.txt` をアクティブノードから共有ディスクの NNM ディレクトリ内の `licenses.txt` にコピーします。
6. HA リソースグループをメンテナンスモードにするページの 99に従って、HA リソースグループをメンテナンスモードにします。

7 NNMi を停止します。

```
ovstop -c
```

8 NNMi 管理サーバーの IP アドレスまたはノード名を変更します。

a ov.conf ファイルの NNM_INTERFACE エントリーを編集して、新しいホスト名または IP アドレスに変更します。

b ovspmd.auth ファイル内の旧ホスト名を含む行を編集して、新しいホスト名を含むようにします。

ov.conf ファイルと ovspmd.auth ファイルは、以下の場所にあります。

- *Windows* の場合：%NnmDataDir%\shared\nnm\conf
- *UNIX* の場合：$NnmDataDir/shared/nnm/conf

9 クラスタ設定情報を設定します。

a NNMi HA リソースグループを停止します。

— *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhastoprg.ovpl NNM
<resource_group>
```

— *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhastoprg.ovpl NNM
<resource_group>
```

b 新しい IP アドレスを使うように、クラスタ設定を変更します。

— *Microsoft Cluster Services* の場合：

クラスタアドミニストレータで、***<resource_group>*** を開きます。

***<resource_group>*-ip** をダブルクリックして、**パラメータ**を選択し、新しい IP アドレスを入力します。

— *HP ServiceGuard* の場合：アクティブな HA クラスタノードで、以下のファイルを編集します。

HP-UX の場合：/etc/cmcluster/*<resource_group>*/*<resource_group>*.cntl

Linux の場合：/usr/local/cmcluster/conf/*<resource_group>*/*<resource_group>*.cntl

IP [0] =*<old_IP_address>* を IP [0] =*<new_IP_address>* で置き換えます。そして、cmapplyconf を使って残りのシステムをすべてアップデートします。

— *VERITAS Cluster Server* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhargconfigure.ovpl NNM
<resource_group> -set_value <resource_group>-ip
Address <new_IP_address>
```

c NNMi HA リソースグループを起動します。

— *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM
<resource_group>
```

— *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM
<resource_group>
```

10 NNMi を正常に起動できたことを確認します。

```
ovstatus -c
```

すべての NNMi サービスで、[実行中] 状態が表示される必要があります。

11 HA リソースグループのメンテナンスモードを解除するページの 99に従って、HA リソースグループのメンテナンスモードを解除します。

フェイルオーバーを行わせないように NNMi を停止する

NNMi のメンテナンスを行う必要がある場合は、アクティブクラスタノードの NNMi を、パッシブノードへフェイルオーバーさせないように停止できます。アクティブクラスタノードで以下の手順を実行します。

1 HA リソースグループをメンテナンスモードにするページの 99に従って、HA リソースグループをメンテナンスモードにします。

2 NNMi を停止します。

```
ovstop -c
```

メンテナンス後に NNMi を再起動する

フェイルオーバーしないように NNMi を停止した場合は、以下の手順を実行して、NNMi と HA モニタリングを再起動します。

1 NNMi を起動します。

```
ovstart -c
```

2 NNMi を正常に起動できたことを確認します。

```
ovstatus -c
```

すべての NNMi サービスで、[実行中] 状態が表示される必要があります。

3 HA リソースグループのメンテナンスモードを解除するページの 99に従って、HA リソースグループのメンテナンスモードを解除します。

NNMi HA クラスタ内のアドオン iSPI のメンテナンス

iSPI は、NNMi に密接にリンクしています。アドオン iSPI を NNMi HA クラスタ内のノードにインストールする場合は、NNMi HA クラスタのメンテナンス手順を使います。

HA クラスタ内の専用サーバー上の NNM iSPI for Performance のメンテナンス

NNM iSPI for Performance の起動と停止

NNM iSPI for Performance を HA 下で実行している場合は、HA のメンテナンスが目的の指示がない限り、`startSPI`、`startERS`、`stopSPI`、`stopERS` などのコマンドは、*使わない*でください。また、NNM iSPI for Performance ユーザーインタフェースの **Start** ボタンや **Stop** ボタンも*使わない*でください。通常のオペレーションでは、NNMi に用意されている HA コマンドまたは HA 製品の適切なコマンドを使って、リソースグループの起動や停止を行います。

クラスタ環境で NNM iSPI for Performance システムを変更する

クラスタ環境で NNMi のホスト名や IP アドレスを変更するページの 99 の手順に従って、NNM iSPI for Performance の使用法をチューニングします。

フェイルオーバーを行わせないように NNM iSPI for Performance を停止する

NNM iSPI for Performance のメンテナンスを行う必要がある場合は、アクティブクラスタノードの NNM iSPI for Performance を、パッシブノードへフェイルオーバーさせないように停止できます。アクティブノードで以下の手順を実行します。

1 HA リソースグループをメンテナンスモードにするページの 99に従って、HA リソースグループをメンテナンスモードにします。

   ファイルは空で構いません。

2 NNM iSPI for Performance を停止します。

   - *Windows* の場合：

     ```
     %NnmInstallDir%\NNMPerformanceSPI\bin\stopSPI.ovpl
     %NnmInstallDir%\NNMPerformanceSPI\bin\stopERS.ovpl
     ```

   - *UNIX* の場合：

     ```
     $NnmInstallDir/NNMPerformanceSPI/bin/stopSPI.ovpl
     $NnmInstallDir/NNMPerformanceSPI/bin/stopERS.ovpl
     ```

メンテナンス後に NNM iSPI for Performance を再起動する

フェイルオーバーしないように NNM iSPI for Performance を停止した場合は、以下の手順を実行して、NNM iSPI for Performance と HA モニタリングを再起動します。

1 NNM iSPI for Performance を起動します。

   - *Windows* の場合：

     ```
     %NnmInstallDir%\NNMPerformanceSPI\bin\startSPI.ovpl
     %NnmInstallDir%\NNMPerformanceSPI\bin\startERS.ovpl
     ```

   - *UNIX* の場合：

     ```
     $NnmInstallDir/NNMPerformanceSPI/bin/startSPI.ovpl
     $NnmInstallDir/NNMPerformanceSPI/bin/startERS.ovpl
     ```

2 NNM iSPI for Performance を正常に起動できたことを確認します。

   - *Windows* の場合：

     ```
     %NnmInstallDir%\NNMPerformanceSPI\bin\statusSPI.ovpl
     %NnmInstallDir%\NNMPerformanceSPI\bin\statusERS.ovpl
     ```

   - *UNIX* の場合：

     ```
     $NnmInstallDir/NNMPerformanceSPI/bin/statusSPI.ovpl
     $NnmInstallDir/NNMPerformanceSPI/bin/statusERS.ovpl
     ```

3 HA リソースグループのメンテナンスモードを解除するページの 99に従って、HA リソースグループのメンテナンスモードを解除します。

HA の設定解除

HA クラスタ内の NNMi の設定を解除する

NNMi ノードを HA クラスタから削除する手順には、NNMi のインスタンスの HA 設定を解除する手順も含まれます。設定を解除すると、NNMi のインスタンスをスタンドアロン管理サーバーとして実行できます。また、そのノードから NNMi をアンインストールできます。

高可用性用の NNMi の設定を維持するには、HA クラスタに、NNMi を実行中の 1 つのノードと、少なくとも、1 つのパッシブ NNMi ノードが必要です。HA クラスタから NNMi を完全に削除するには、クラスタ内のすべてのノードで HA 機能の設定を解除します。

HA クラスタの NNMi の設定を完全に解除するには、以下の手順に従います。

1 HA クラスタ内のアクティブなノードを特定します。

- *Windows* の場合：

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

- *UNIX* の場合：

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

2 各パッシブノードで、HA クラスタからアドオン iSPI の設定を解除します。各 iSPI に対して、次のコマンドを実行します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM
-addon <iSPI_PM_Name>
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM
-addon <iSPI_PM_Name>
```

`<iSPI_PM_Name>` には、iSPI が NNMi 管理サーバーにインストールする Perl モジュールのベース名を指定します。`<iSPI_PM_Name>` に指定する値を調べるには、89 ページの表 6、または統合とプラグイン章の該当する iSPI の項を参照してください。

3 各パッシブノードで、HA クラスタから NNMi の設定を解除します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM
<resource_group>
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM
<resource_group>
```

このコマンドによって、共有ディスクへのアクセス権は失われますが、ディスクグループやボリュームグループの設定が解除されるわけではありません。

4 各パッシブノードで、リソースグループ固有のファイルを削除します。

- *Windows* の場合：

  Windows Explorer で、`%NnmDataDir%\hacluster\<resource_group>\` フォルダー内のすべてのファイルを削除します。

- *HP-UX* の場合：

  ```
  rm -rf $NnmDataDir/hacluster/<resource_group>/*
  rm -rf /etc/cmcluster/<resource_group>/*
  ```

- *Linux* の場合：

  ```
  rm -rf $NnmDataDir/hacluster/<resource_group>/*
  rm -rf /usr/local/cmcluster/conf/<resource_group>/*
  ```

- *Solaris* の場合：

  ```
  rm -rf $NnmDataDir/hacluster/<resource_group>/*
  ```

高可用性用の NNMi の設定を維持する場合は、以上の手順で終了です。

5 アクティブノードで、HA クラスタからアドオン iSPI の設定を解除します。各 iSPI に対して、次のコマンドを実行します。

- *Windows* の場合：

  ```
  %NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM
  -addon <iSPI_PM_Name>
  ```

- *UNIX* の場合：

  ```
  $NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM
  -addon <iSPI_PM_Name>
  ```

6 アクティブノードで、NNMi HA リソースグループを停止します。

- *Windows* の場合：

  ```
  %NnmInstallDir%\misc\nnm\ha\nnmhastoprg.ovpl NNM
  <resource_group>
  ```

- *UNIX* の場合：

  ```
  $NnmInstallDir/misc/nnm/ha/nnmhastoprg.ovpl NNM
  <resource_group>
  ```

このコマンドでは、共有ディスクへのアクセス権は削除しません。また、ディスクグループやボリュームグループの設定も解除しません。

7 アクティブノードで、リソースグループ固有のファイルを削除します。

- *Windows* の場合：

  Windows Explorer で、`%NnmDataDir%\hacluster\<resource_group>\` フォルダー内のすべてのファイルを削除します。

- *HP-UX* の場合：

  ```
  rm -rf $NnmDataDir/hacluster/<resource_group>/*
  rm -rf /etc/cmcluster/<resource_group>/*
  ```

- *Linux* の場合：

  ```
  rm -rf $NnmDataDir/hacluster/<resource_group>/*
  rm -rf /usr/local/cmcluster/conf/<resource_group>/*
  ```

- *Solaris* の場合：

  ```
  rm -rf $NnmDataDir/hacluster/<resource_group>/*
  ```

8 共有ディスクをマウント解除します。

- NNMi HA クラスタの再設定を予定している場合は、ディスクを現状のままで保管できます。
- 共有ディスクを別の目的で使う場合は、保管するデータをすべてコピーして（次の手順で説明）から、HA 製品のコマンドを使って、ディスクグループとボリュームグループの設定を解除します。

NNMi を HA の外部の任意のノードで既存のデータベースを使って実行する場合は、以下の手順を実行します。

1 アクティブなノードで（存在する場合）、NNMi が実行中ではないことを確認します。

```
ovstop
```

あるいは、タスクマネージャ (Windows) または ps コマンド (UNIX) を使って、ovspmd プロセスのステータスをチェックします。

2 現在のノード（HA の外部で NNMi の実行を予定しているノード）で、NNMi が実行中ではないことを確認します。

```
ovstop
```

データの破壊を避けるために、NNMi または NNM iSPI for Performance のインスタンスが動作中ではないことや、共有ディスクにアクセス中ではないことを確認します。

3 適切なオペレーティングシステムのコマンドを使って、共有ディスクをマウントします。例：

- *Windows* の場合：Windows Explorer を使います。
- *UNIX* の場合：

  ```
  vgchange -a e vgnnm
  mount /dev/vgnnm/lvnnm /nnmmount
  ```

4 共有ディスクからノードに、NNMi ファイルをコピーします。

- *Windows* の場合：

  ```
  %NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM
  -from <HA_mount_point>
  ```

- *UNIX* の場合：

  ```
  $NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM
  -from <HA_mount_point>
  ```

5 適切なオペレーティングシステムのコマンドを使って、共有ディスクのマウントを解除します。例：

- *Windows* の場合：Windows Explorer とディスクの管理を使います。
- *UNIX* の場合：

  ```
  umount /nnmmount
  vgchange -a n vgnnm
  ```

6 NNMi を起動します。

```
ovstart -c
```

従来、NNMi HA リソースグループで使われていたデータベースのコピーを使って、NNMi が起動されます。この NNMi 管理サーバーから管理対象としないノードの NNMi 設定を手動で削除します。

HA クラスタ内の NNM iSPI for Performance の設定を解除する

NNM iSPI for Performance ノードを HA クラスタから削除する手順には、iSPI のインスタンスの HA 設定を解除する手順も含まれます。設定を解除すると、iSPI のインスタンスをスタンドアロンサーバーとして実行できます。また、そのノードから iSPI をアンインストールできます。

高可用性用の NNM iSPI for Performance の設定を維持するには、HA クラスタに、iSPI を実行中の 1 つのノードと、少なくとも、1 つのパッシブ iSPI ノードが必要です。HA クラスタから NNM iSPI for Performance を完全に削除するには、クラスタ内のすべてのノードで HA 機能の設定を解除します。

HA クラスタの NNM iSPI for Performance の設定を完全に解除するには、以下の手順に従います。

1 HA クラスタ内のアクティブなノードを特定します。

- *Windows* の場合：

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

- *UNIX* の場合：

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

2 各パッシブノードで、HA クラスタから NNM iSPI for Performance の設定を解除します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl PerfSPIHA
<resource_group>
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl PerfSPIHA
<resource_group>
```

このコマンドによって、共有ディスクへのアクセス権は失われますが、ディスクグループやボリュームグループの設定が解除されるわけではありません。

3 各パッシブノードで、リソースグループ固有のファイルを削除します。

- *Windows* の場合：

  Windows Explorer で、`%NnmDataDir%\hacluster\<resource_group>\` フォルダー内のすべてのファイルを削除します。

- *HP-UX* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
rm -rf /etc/cmcluster/<resource_group>/*
```

- *Linux* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
rm -rf /usr/local/cmcluster/conf/<resource_group>/*
```

- *Solaris* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
```

高可用性用の NNM iSPI for Performance の設定を維持する場合は、以上の手順で終了です。

4 アクティブノードで、パフォーマンス HA リソースグループを停止します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhastoprg.ovpl PerfSPIHA
<resource_group>
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhastoprg.ovpl PerfSPIHA
<resource_group>
```

このコマンドでは、共有ディスクへのアクセス権は削除しません。また、ディスクグループやボリュームグループの設定も解除しません。

5 アクティブノードで、リソースグループ固有のファイルを削除します。

- *Windows* の場合：

  Windows Explorer で、`%NnmDataDir%\hacluster\<resource_group>\` フォルダー内のすべてのファイルを削除します。

- *HP-UX* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
rm -rf /etc/cmcluster/<resource_group>/*
```

- *Linux* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
rm -rf /usr/local/cmcluster/conf/<resource_group>/*
```

- *Solaris* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
```

6 共有ディスクをマウント解除します。

- パフォーマンス HA クラスタの再設定を予定している場合は、ディスクを現状のままで保管できます。
- 共有ディスクを別の目的で使う場合は、保管するデータをすべてコピーして（次の手順で説明）から、HA 製品のコマンドを使って、ディスクグループとボリュームグループの設定を解除します。

NNM iSPI for Performance を HA の外部の任意のノードで既存のデータベースを使って実行する場合は、以下の手順を実行します。

1 アクティブなノードで、NNM iSPI for Performance が実行中ではないことを確認します。

- *Windows* の場合：

```
%NnmInstallDir%\NNMPerformanceSPI\bin\stopSPI.ovpl
%NnmInstallDir%\NNMPerformanceSPI\bin\stopERS.ovpl
```

- *UNIX* の場合：

  $NnmInstallDir/NNMPerformanceSPI/bin/stopSPI.ovpl
  $NnmInstallDir/NNMPerformanceSPI/bin/stopERS.ovpl

2 現在のノード（HA の外部で NNM iSPI for Performance の実行を予定しているノード）で、iSPI が実行中ではないことを確認します。

- *Windows* の場合：

  %NnmInstallDir%\NNMPerformanceSPI\bin\stopSPI.ovpl
  %NnmInstallDir%\NNMPerformanceSPI\bin\stopERS.ovpl

- *UNIX* の場合：

  $NnmInstallDir/NNMPerformanceSPI/bin/stopSPI.ovpl
  $NnmInstallDir/NNMPerformanceSPI/bin/stopERS.ovpl

データの破壊を避けるために、iSPI のインスタンスが動作中ではないことや、共有ディスクにアクセス中ではないことを確認します。

3 適切なオペレーティングシステムのコマンドを使って、共有ディスクをマウントします。例：

- *Windows* の場合：Windows Explorer を使います。
- *UNIX* の場合：

```
vgchange -a e vgnnm
mount /dev/vgnnm/lvnnm /nnmmount
```

4 共有ディスクからノードに、iSPI ファイルをコピーします。

- *Windows* の場合：

  %NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl PerfSPIHA -from <*HA_mount_point*>

- *UNIX* の場合：

  $NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl PerfSPIHA -from <*HA_mount_point*>

5 適切なオペレーティングシステムのコマンドを使って、共有ディスクのマウントを解除します。例：

- *Windows* の場合：Windows Explorer とディスクの管理を使います。
- *UNIX* の場合：

```
umount /nnmmount
vgchange -a n vgnnm
```

6 NNM iSPI for Performance を起動します。

- *Windows* の場合：

  %NnmInstallDir%\NNMPerformanceSPI\bin\startSPI.ovpl
  %NnmInstallDir%\NNMPerformanceSPI\bin\startERS.ovpl

- *UNIX* の場合：

  $NnmInstallDir/NNMPerformanceSPI/bin/startSPI.ovpl
  $NnmInstallDir/NNMPerformanceSPI/bin/startERS.ovpl

従来、パフォーマンス HA リソースグループで使われていたデータベースのコピーを使って、NNM iSPI for Performance が起動されます。

HA 下の NNMi を NNMi 8.0x から NNMi 8.10 にアップグレードする

HA 下の NNMi 8.0x から NNMi 8.10 にアップグレードするには、HA の NNMi 8.0x の設定を解除して、NNMi 8.10 をインストールし、アップデートされたデータベーススキーマを取得して、HA 用に NNMi を再設定します。以下の手順に従ってください。

1 すべての HA ノードで、NNMi 8.0x の設定が同じであることを確認します。

- *Windows* の場合：

  %NnmInstallDir%\misc\nnm\ha\nnmdatareplicator.ovpl NNM

- *UNIX* の場合：

  $NnmInstallDir/misc/nnm/ha/nnmdatareplicator.ovpl NNM

2 NNMi 8.0x HA クラスタ内のアクティブなノードを特定します。

- *Windows* の場合：

  %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl -group *<resource_group>* -state

- *UNIX* の場合：

  $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -group *<resource_group>* -state

3 各パッシブノードで、NNMi 8.0x の設定を解除します。

a HA クラスタの NNMi の設定を解除します。

— *Windows* の場合：

%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl *<resource_group>*

— *UNIX* の場合：

$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl *<resource_group>*

このコマンドによって、共有ディスクへのアクセス権は失われますが、ディスクグループやボリュームグループの設定が解除されるわけではありません。

b リソースグループ固有のファイルを削除します。

— *Windows* の場合：

Windows Explorer で、%NnmDataDir%\hacluster*<resource_group>*\ フォルダー内のすべてのファイルを削除します。

— *HP-UX* の場合：

rm -rf $NnmDataDir/hacluster/*<resource_group>*/*
rm -rf /etc/cmcluster/*<resource_group>*/*

— *Linux* の場合：

rm -rf $NnmDataDir/hacluster/*<resource_group>*/*
rm -rf /usr/local/cmcluster/conf/*<resource_group>*/*

— *Solaris* の場合：

rm -rf $NnmDataDir/hacluster/*<resource_group>*/*

4 アクティブノードで、NNMi 8.0x の設定を解除します。

a 以下のメンテナンスファイルを作成して、HA リソースグループのモニタリングを無効にします。

— *Windows* の場合：`%NnmDataDir%\hacluster\maint_NNM`

— *UNIX* の場合：
`$NnmDataDir/hacluster/<resource_group>/maint_NNM`

ファイルは空で構いません。

b NNMi を停止します。

```
ovstop -c
```

データの破壊を避けるために、NNMi のインスタンスが動作中ではないことや、共有ディスクにアクセス中ではないことを確認します。

c 共有ディスク上の NNMi データベースをバックアップします。

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

このコマンドの詳細については、NNMi のバックアップおよびリストアツールページの 123 を参照してください。

d 共有ディスクからノードに、NNMi ファイルをコピーします。

— *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl
-from <HA_mount_point>
```

— *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl
-from <HA_mount_point>
```

e NNMi のファイルとディレクトリをすべて共有ディスクから削除します。

— *Windows* の場合：Windows Explorer を使って、共有ディスクのマウントポイント（`%HA_MOUNT_POINT%`、たとえば、`S:\`）下のファイルをすべて削除します。

— *UNIX* の場合：

```
rm -rf $HA_MOUNT_POINT/*
```

f NNMi HA リソースグループを停止します。

— *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhastoprg.ovpl NNM
<resource_group>
```

— *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhastoprg.ovpl NNM
<resource_group>
```

g HA クラスタの NNMi の設定を解除します。

— *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl
<resource_group>
```

— *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl
<resource_group>
```

このコマンドによって、共有ディスクへのアクセス権は失われますが、ディスクグループやボリュームグループの設定が解除されるわけではありません。

h リソースグループ固有のファイルを削除します。

— *Windows* の場合：

Windows Explorer で、`%NnmDataDir%\hacluster\<resource_group>\` フォルダー内のすべてのファイルを削除します。

— *HP-UX* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
rm -rf /etc/cmcluster/<resource_group>/*
```

— *Linux* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
rm -rf /usr/local/cmcluster/conf/<resource_group>/*
```

— *Solaris* の場合：

```
rm -rf $NnmDataDir/hacluster/<resource_group>/*
```

i メンテナンス ファイルを削除します。

— *Windows* の場合：`%NnmDataDir%\hacluster\maint_NNM`

— *UNIX* の場合：
`$NnmDataDir/hacluster/<resource_group>/maint_NNM`

5 HA から NNMi 8.0x の設定を解除する前はアクティブだったノードで、NNMi 8.10 を設定します。

a NNMi 8.10 アップグレードをインストールします。

アップグレードプロセスが NNMi を起動し、NNMi データベーススキーマがアップデートされます。

b プライマリクラスタノードでの NNMi の設定ページの 86 の手順に従ってください。

以下の作業は不要です。

— ディスクデバイスグループと論理ボリュームの定義

— 共有ディスクのマウントポイントの作成

— 共有ディスクの設定

6 HA から NNMi 8.0x の設定を解除する前はパッシブだった各ノードで、NNMi 8.10 を設定します。

a NNMi 8.10 アップグレードをインストールします。

b 最初のセカンダリノードで、セカンダリクラスタノードでの NNMi の設定ページの 88 の手順を実行します。

共有ディスクのマウントポイントの作成は不要です。

HA 設定のトラブルシューティング

一般的な HA のトラブルシューティング

ここでの内容は、すべての HP NNM i-series Software 製品に適用されます。

エラー：Wrong Number of Arguments（引き数の数が間違っています）

NNMi 8.10 から、Perl モジュール製品の名前は、大部分の NNMi HA 設定コマンドで必須パラメータになりました。

- NNMi の場合は、値 `NNM` を使います。
- iSPI で使う値を調べるには、89 ページの表 6、または統合とプラグイン章の該当する iSPI の項を参照してください。

製品スタートアップのタイムアウト (Solaris)

1 つ以上の `/var/adm/messages*` ファイルに、次の例のようなメッセージが含まれます。

```
VCS ERROR V-16-1-13012 Thread() Resource(<resource group>-app):
online procedure did not complete within the expected time.
```

このメッセージは、製品が Veritas タイムアウト値の範囲内で完全には起動できなかったことを示しています。NNMi に付属した HA 設定スクリプトでは、タイムアウトは 15 分と定義されています。

Solaris オペレーティングシステムでの Veritas タイムアウト値を変更するには、以下のコマンドを、以下の順番で、実行します。

```
/opt/VRTSvcs/bin/haconf makerw
/opt/VRTSvcs/bin/hares modify <resource_group>-app OnlineTimeout <value in seconds>
/opt/VRTSvcs/bin/haconf dump makero
```

アクティブなクラスタノードのログファイルが更新されない

これは正常です。ログファイルは、共有ディスクにリダイレクトされているため、このような状況になります。

NNMi の場合は、`ov.conf` ファイル内の `HA_NNM_LOG_DIR` で指定された場所にあるログファイルを調べてください。

NNM iSPI for Performance の場合は、ローカルログファイルは HA の設定中に削除されます。共有ディスクの iSPI ログファイルの場所は、NNM iSPI for Performance HA ログファイルページの 121 を参照してください。

HA リソースグループが特定のクラスタノードでは起動できない

nnmhargconfigure.ovpl コマンドで NNMi HA リソースグループを正常に起動/停止/切り替えできない場合は、次の情報を調べてください。

- *Windows* の場合：
  - クラスタアドミニストレータで、リソースグループおよびそれを構成するリソースの状態を調べてください。
  - イベントビューアのログにエラーが記録されていないか調べてください。
- *HP-UX* の場合：

  /etc/cmcluster/*<resource_group>*/*<resource_group>*.cntl.log ファイルと syslog ファイルにエラーが記録されていないか調べてください。良くある原因は、リソースを追加できない状態（たとえば、ディスクグループの設定を誤っているため、アクティブにできない）のままで、システムが放置されていることです。
- *Linux* の場合：

  /usr/local/cmcluster/conf/*<resource_group>*/*<resource_group>*.cntl.log ファイルと syslog ファイルにエラーが記録されていないか調べてください。良くある原因は、リソースを追加できない状態（たとえば、ディスクグループの設定を誤っているため、アクティブにできない）のままで、システムが放置されていることです。
- *Solaris* の場合：
  - **/opt/VRTSvcs/bin/hares -state** を実行して、リソースの状態を調べます。
  - 障害が発生しているリソースでは、障害が発生しているリソース用の /var/VRTSvcs/log/*<resource>*.log ファイルを調べます。リソースは、IP*.log、Mount*.log、Volume*.log などのエージェントタイプで指定します。

原因となっているリソースを特定できない場合は、HA 製品のコマンドを使って、HA リソースグループを手動で起動します。

1. 共有ディスクをマウントします。
2. ネットワークインタフェースに仮想ホストを割り当てます。
   - *Windows* の場合：
     - クラスタアドミニストレータを起動します。
     - リソースグループを展開します。
     - ［***<resource_group>*-ip**］を右クリックして、［**Bring Online**］をクリックします。
   - *HP-UX* の場合：/usr/sbin/cmmodnet を実行して、IP アドレスを追加します。
   - *Linux* の場合：/usr/local/cmcluster/bin/cmmodnet を実行して、IP アドレスを追加します。
   - *Solaris* の場合：
     /opt/VRTSvcs/bin/hares -online *<resource_group>*-ip -sys *<local_hostname>*

3 HA リソースグループを起動します。例：

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhastart.ovpl NNM
-start <resource_group>
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhastart.ovpl NNM
-start <resource_group>
```

リターンコード 0 は、NNMi を正常に起動できたことを意味します。

リターンコード 1 は、NNMi を正常に起動できなかったことを意味します。

NNMi 固有の HA のトラブルシューティング

この項の内容が適用されるのは、NNMi のみの HA 設定です。

すべてのクラスタノードの設定解除後に、HA 用の NNMi を再び有効にする

すべての NNMi HA クラスタノードの設定を解除した場合は、NNMi の共有ディスクのマウントポイントへのリンクが、ov.conf ファイルから削除されます。共有ディスク内のデータを上書きすることなく、マウントポイントへのリンクを作成しなおすには、プライマリノードで以下の手順を実行します。

1 NNMi が実行中であれば、停止します。

```
ovstop -c
```

2 共有ディスクへのリンクを削除します。

- *Windows* の場合：

```
%NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM
-setmount <HA_mount_point>
```

- *UNIX* の場合：

```
$NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM
-setmount <HA_mount_point>
```

3 ov.conf ファイルの HA マウントポイント関連のエントリーを確認します。

ov.conf ファイルの場所は、NNMi HA 設定ファイルページの 118 を参照してください。

NNMi を HA 下で正常に起動できない

以下のいずれかの状況が発生しています。

- NNMi を HA 用に設定した後、ovspmd を起動できません。(ovstatus では、ovspmd が起動されていませんと表示されます。)
- ovstart または ovstop から、以下のメッセージが返されます。

```
ovstart: Must specify managers by name if running on OVW
Client
```

NNMi を HA 下で実行している場合は、HA のメンテナンスが目的の指示がない限り、ovstart コマンドや ovstop コマンドは、*使わない*でください。

NNMi の設定で、NNMi を実行中のシステム以外のシステムを指しています。この問題を解決するには、ov.conf ファイルに以下の項目に対応した適切なエントリーがあるか確認します。

- 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>

ov.conf ファイルの場所は、NNMi HA 設定ファイルページの 118を参照してください。

HA の設定後、nmsdbmgr を起動できない

この状況は、通常、-to オプションを付けた nnmhadisk.ovpl コマンドを実行した直後に NNMi を起動した場合に発生します。（nnmhaconfigure.ovpl コマンドが実行されていません。）この状況では、ov.conf ファイルの HA_POSTGRES_DIR エントリーは、共有ディスクの組み込みデータベースの場所を指していますが、この場所は NNMi からはアクセスできません。

nnmhaconfigure.ovpl を実行してから、HA 設定を完了させます。

HA の設定後、pmd を起動できない

この状況は、通常、共有ディスクを正しく設定しなかったなどの設定エラー後に発生します。pmd プロセスの障害は、ovjboss プロセスを完全に起動できなかった場合に発生します。

以下のログファイルを調べてください。

- *Windows* の場合：
  %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\jbossServer.log
- *UNIX* の場合：$HA_MOUNT_POINT/NNM/dataDir/log/nnm/jbossServer.log

ディスクフェイルオーバーが行われない

この状況は、該当するプラットフォームで共有ディスクがサポートされていない場合に発生する可能性があります。HA 製品、オペレーティングシステム、ディスクのメーカーのマニュアルなどを見て、これらの製品を混在させて使うことができるか確認してください。

ディスクで障害が発生した場合は、NNMi はフェイルオーバーによって起動されません。nmsdbmgr が失敗する理由の多くは、HA_POSTGRES_DIR ディレクトリが存在しないことです。共有ディスクがマウント済みであり、該当するファイルにアクセスできる状態になっていることを確認してください。

フェイルオーバー後にセカンダリノードで共有ディスクファイルが見つからない

この状況は、通常、共有ディスクがマウントされていないときに、-to オプションを付けた nnmhadisk.ovpl コマンドを実行した場合に発生します。この場合には、データファイルはローカルディスクにコピーされ、共有ディスクには格納されません。

iSPI 固有の HA のトラブルシューティング

この項の内容は、すべての HP NNM i-series Software iSPI のアドオン HA 設定に適用されます。

iSPI を HA 下で正常に起動できない

以下の手順を開始する前に、当社サポート担当に支援を要請してください。

1 問題となっているクラスタノードで、NNM_ADD_ON_PRODUCTS 属性の現在の値を調べて、各 iSPI が正しく登録されているか確認します。
   - *Windows* の場合：以下のコマンドを実行します。

     cluster group *<resource_group>* /PRIV

     出力内の PUBLIC.NNM_ADD_ON_PRODUCTS を探します。
   - *HP-UX* の場合：/etc/cmcluster/*<resource_group>*/*<resource_group>*.public.env ファイルを調べます。
   - *Linux* の場合：/usr/local/cmcluster/conf/*<resource_group>*/*<resource_group>*.public.env ファイルを調べます。
   - *Solaris* の場合：UserStrGlobal 内にリソースグループ指定の一部として格納されています。
2 NNM_ADD_ON_PRODUCTS の値が正しくない場合は、その値を変更します。
   - *Windows* の場合：以下のコマンドを実行します。

     cluster group *<resource_group>* /PRIV
     <*attribute_name*>=<*value*>

Windows レジストリは直接編集しないでください。

   - *HP-UX* の場合：他のクラスタノードにある、正しい /etc/cmcluster/*<resource_group>*/*<resource_group>*.public.env ファイルをこのクラスタノードにコピーします。
   - *Linux* の場合：他のクラスタノードにある、正しい /usr/local/cmcluster/conf/*<resource_group>*/*<resource_group>*.public.env ファイルをこのクラスタ ノードにコピーします。
   - *Solaris* の場合：UserStrGlobal をアップデートします。

     /opt/VRTSvcs/bin/haconf -makerw
     /opt/VRTSvcs/bin/hagrp -modify *<resource_group>*
     UserStrGlobal "*<newvalue>*"

HA 設定リファレンス

NNMi HA 設定ファイル

表 8 に、NNMi HA 設定ファイルを示します。これらのファイルは、NNMi と NNMi 管理サーバー上のアドオンの iSPI に適用されます。これらのファイルは、以下の場所にインストールされます。

- *Windows* の場合：`%NnmDataDir%\shared\nnm\conf`
- *UNIX* の場合：`$NnmDataDir/shared/nnm/conf`

表 8　NNMi HA 設定ファイル

ファイル名	説明
ov.conf	このファイルは、NNMi HA 実装の状態を示し、`nnmhaclusterinfo.ovpl` コマンドによって更新されます。NNMi の各プロセスは、このファイルを読み取って、HA 設定を確認します。
nnmdatareplicator.conf	このファイルは、`nnmdatareplicator.ovpl` コマンドで、アクティブノードからパッシブノードへのデータレプリケーションに含む NNMi のフォルダーとファイルを調べるために使われます。NNMi 設定のレプリケーション用に異なる手段を実装する場合は、含めるデータのリストは、このファイルを参照してください。 詳細については、このファイルのコメントを参照してください。

NNM iSPI for Performance HA 設定ファイル

アドオンの NNM iSPI for Performance は、HA を設定用に `ov.conf` ファイルを使います。

専用サーバー上の NNM iSPI for Performance は、HA 設定用に以下のファイルを使います。

- *Windows* の場合：
  - `%NnmDataDir%\shared\perfSpi\conf\perfspi.conf`
- *UNIX* の場合：
  - `$NnmDataDir/shared/perfSpi/conf/perfspi.conf`

NNMi に付属している HA 設定スクリプト

表 9 に、NNMi に付属している HA 設定スクリプトを示します。NNMi に付属しているスクリプトは、カスタマ Perl モジュールを持つすべての製品に HA を設定する場合に使うことができる便利なスクリプトです。必要に応じて、HA 製品に付属しているコマンドを使って、NNMi 用に HA を設定できます。

NNMi 管理サーバーでは、NNMi に付属している HA 設定スクリプトは、以下の場所にインストールされます。

- *Windows* の場合：`%NnmInstallDir%\misc\nnm\ha`
- *UNIX* の場合：`$NnmInstallDir/misc/nnm/ha`

HP-UX オペレーティングシステムと Solaris オペレーティングシステムでは、これらのディレクトリに廃止されたスクリプト（ov* という名前を持つ）が含まれています。これらのスクリプトは実行しないでください。

表 9　NNMi HA 設定スクリプト

スクリプト名	説明
nnmhaconfigure.ovpl	NNMi または iSPI を HA クラスタ用に設定します。 このスクリプトは、HA クラスタ内のすべてのノードで実行してください。
nnmhaunconfigure.ovpl	HA クラスタの NNMi または iSPI の設定を解除します。 必要に応じて、HA クラスタ内の 1 つ以上のノードでこのスクリプトを実行します。
nnmhaclusterinfo.ovpl	NNMi に関するクラスタ情報を取得します。 このスクリプトは、必要に応じて、HA クラスタ内の任意のノードで実行します。
nnmhadisk.ovpl	データファイルを、NNMi および iSPI と共有ディスクの間でコピーします。 HA の設定時には、このスクリプトはプライマリノードで実行します。 それ以外の場合は、この章の手順に従って、このスクリプトを実行します。
nnmhastartrg.ovpl	HA クラスタで NNMi を起動します。 HA の設定時には、このスクリプトはプライマリノードで実行します。
nnmdatareplicator.ovpl	`nnmdatareplicator.conf` 設定ファイルを調べて、リモートシステムに送信するファイルの変更やコピーを確認します。
nnmharg.ovpl	HA クラスタの NNMi を起動 / 停止 / 監視します。 ServiceGuard 設定の場合に、*<resource_group>*`.cntl` によって使われます（HP-UX と Linux）。 Solaris では、Veritas の起動 / 停止 / 監視スクリプトで使われます。 （`nnmhargconfigure.ovpl` でこのスクリプトの使用法が設定されます。） また、トレースを有効 / 無効にするために、`nnmhastartrg.ovpl` でも使われます。
nnmhargconfigure.ovpl	HA のリソースとリソースグループを設定します。`nnmhaconfigure.ovpl` と `nnmhaunconfigure.ovpl` で使われます。
nnmhastart.ovpl	HA クラスタで NNMi を起動します。`nnmharg.ovpl` で使われます。
nnmhastop.ovpl	HA クラスタの NNMi を停止します。`nnmharg.ovpl` で使われます。
nnmhamonitor.ovpl	HA クラスタの NNMi プロセスを監視します。`nnmharg.ovpl` で使われます。
nnmhamscs.vbs	Microsoft Cluster Services の HA クラスタで、NNMi プロセスを起動 / 停止 / 監視するスクリプトを作成するためのテンプレートです。生成されるスクリプトは、次の場所に格納され、Microsoft Cluster Services で使われます。 `%NnmDataDir%\hacluster\`*<resource_group>*`\hamscs.vbs`

NNMi HA 設定のログファイル

以下のログファイルは、NNMi と NNMi 管理サーバー上のアドオン iSPI 用の HA 設定に適用されます。

- *Windows* 設定：
  - `%NnmDataDir%\tmp\HA_nnmhaserver.log`
  - `%NnmDataDir%\log\haconfigure.log`
- *UNIX* 設定：
  - `$NnmDataDir/tmp/HA_nnmhaserver.log`
  - `$NnmDataDir/log/haconfigure.log`
- *Windows* 実行時：
  - イベントビューアのログ
  - `%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`
  - `%HA_MOUNT_POINT%\NNM\dataDir\log\nnm\ovet*.log`
  - `%SystemRoot%\Cluster\cluster.log`
    これは、リソースとリソースグループの追加 / 削除、他の設定上の問題点、起動 / 停止上の問題点を含む、クラスタ実行時の問題点に関するログファイルです。
- *HP-UX* 実行時：
  - `/etc/cmcluster/<resource_group>/<resource_group>.cntl.log`
    これは、リソースグループ用のログファイルです。
  - `/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`
  - `$HA_MOUNT_POINT/NNM/dataDir/log/nnm/ovet*.log`
- *Linux* 実行時：
  - `/usr/local/cmcluster/conf/<resource_group>/<resource_group>.cntl.log`
    これは、リソースグループ用のログファイルです。
  - `/var/log/cmcluster`
    これは、クラスタの問題点に関するログファイルです。
  - `/var/log/messages*`
    これらは、リソースの問題点（ディスクと IP アドレス）に関するログ ファイルです。
  - `$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

— $HA_MOUNT_POINT/NNM/dataDir/log/nnm/ovet*.log

- *Solaris* 実行時：

リソース	ログファイル
<resource_group>-app	• /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 • $HA_MOUNT_POINT/NNM/dataDir/log/nnm/ovet*.log • /var/adm/messages*
<resource_group>-dg *<resource_group>*-volume *<resource_group>*-mount	• /var/VRTSvcs/log/DiskGroup_A.log • /var/VRTSvcs/log/Volume_A.log • /var/VRTSvcs/log/Mount_A.log • /var/adm/messages*
<resource_group>-ip	• /var/VRTSvcs/log/IP_A.log • /var/adm/messages*
オペレーティング システム固有の HA リソース関連の問題は、/var/adm/messages* ファイルを調べてください。*<resource_group>*-app では、プロセスを起動できなかったことに関するメッセージを探してください。	

NNM iSPI for Performance HA ログファイル

パフォーマンス HA リソースグループは、NNMi HA 設定のログファイルページの 120 にリストされているログファイルの多くを使用します。また、この設定では、以下のログファイルを使います。

- *Windows* 設定：
  — %NnmDataDir%\NNMPerformanceSPI\logs\prspiHA.log
- *UNIX* 設定：
  — $NnmDataDir/NNMPerformanceSPI/logs/prspiHA.log
- *Windows* 実行時：
  — %HA_MOUNT_POINT%\NNMPerformanceSPI\dataDir\NNMPerformanceSPI\logs*.log
- *UNIX* 実行時：
  — $HA_MOUNT_POINT/NNMPerformanceSPI/dataDir/NNMPerformanceSPI/logs/*.log

NNMi のバックアップおよびリストアツール

この章では以下について説明します。

- NNMi データバックアップについて
- NNMi バックアップおよびリストアスクリプトの制限
- NNMi データのバックアップ
- NNMi データのリストア
- 組み込みデータベースのみをバックアップおよびリストアする

NNMi データバックアップについて

正しいバックアップおよびリストア方針は、業務の稼働が中断されないようにするための鍵となります。HP Network Node Manager i Software はネットワーク運用のための重要な資産であり、定期的にバックアップする必要があります。

HP では、すべての NNMi データの完全バックアップを毎週行うことを推奨しています。このバックアップを作成するために NNMi を停止する必要はありません。設定変更を開始する前に、領域を限定したバックアップ（NNMi データのバックアップで説明）を必要に応じて実施してください。このようにすると、設定を変更しても期待した効果が見られない場合、周知の作動設定に戻すことが可能になります。

NNMi インストールに関連した重要データは、以下の 2 種類です。

- ファイルシステム内のファイル
- リレーショナルデータベース（組み込みまたは外部）のデータ

この章では、重要な NNMi ファイルおよびデータをバックアップおよびリストアするために NNMi で装備しているツールについて説明しています。NNMi 組み込みデータベースを使用する場合、これらのツールが NNMi インストールをバックアップおよびリストアする上で十分です。外部データベース（Oracle など）を使用する場合、この資料を読んで既存のデータベースバックアップ手順に追加が必要な NNMi ファイルおよびデータベーステーブルを確認してください。

スクリプト名

NNMi には、NNMi データをバックアップおよびリストアするために以下のスクリプトがあります。

- `nnmbackup.ovpl`— 必要なすべてのファイルシステムデータ（設定情報を含む）と組み込みデータベースに保管されたデータをバックアップします。
- `nnmrestore.ovpl`ó`nnmbackup.ovpl` — スクリプトを使用して作成されたバックアップをリストアします。
- `nnmbackupembdb.ovpl`—NNMi 組み込みデータベース（ファイルシステムデータではない）の完全バックアップを、NNMi の稼働中に作成します。
- `nnmrestoreembdb.ovpl`ó`nnmbackupembdb.ovpl`— スクリプトを使用して作成されたバックアップをリストアします。

iSPI との統合

Smart Plug-in (iSPI) からのデータを NNMi バックアップおよびリストアに含めることができます。特定の iSPI については、このガイドの統合とプラグインセクションの該当する章を参照してください。

NNMi バックアップおよびリストアスクリプトの制限

バックアップシステムとリストアシステムは、同じ設定でなければならない

バックアップスクリプトとリストアスクリプトを使用して、ある NNMi 管理サーバーから他の管理サーバーへデータを転送することができます。以下の項目が両方のシステムで同じである必要があります。

- NNMi のバージョン（パッチを含む）
- OS のタイプとバージョン
- キャラクタセット（言語）

以下の項目は、2 つのシステム間で異なっていてもかまいません。

- ホスト名
- IP アドレス

外部データベースはサポートされない

NNMi バックアップおよびリストアスクリプトは、外部データベース（Oracle など）で保管されたデータと連動しません。外部データベースの保守は、既存のデータベースバックアップおよびリストア手順の一環として扱う必要があります。

NNMi データのバックアップ

NNMi が提供するバックアップスクリプトは、同期された NNMi データを指定されたターゲットディレクトリにコピーします。これで、適切なツールを使用してバックアップのコピーを保存できます。

NNMi のバックアップスクリプトは、2 種類のバックアップをサポートします。

- オンラインバックアップは NNMi の稼働中に行われます。最低でも nmsdbmgr サービスが稼働している必要があります。
- オフラインバックアップは、NNMi が完全に停止している間に行われます。

NNMi のバックアップスクリプトを使用して、NNMi データの完全バックアップを作成したり、一部のデータのみを機能に応じてバックアップできます。部分バックアップは、システム間で情報を移動したり NNMi の現在の状態をアーカイブする場合に便利です。

部分バックアップを使用できるのは、オンラインバックアップの場合のみです。バックアップスクリプトを NNMi が完全に停止している間に実行すると、このスクリプトはバックアップ領域にかかわらず組み込みデータベース内のすべてのデータをコピーします。

バックアップスクリプトの構文を以下に示します。表 10 では、スクリプトのコマンドラインのオプションを説明しています。

```
nnmbackup.ovpl [-type (online|offline)]
  [-scope (config|topology|events|all)] [-force] [-archive]
   -target <ディレクトリ>]
```

表 10　nnmbackup.ovpl コマンドラインのオプション

オプション	説明
-type (online|offline)	バックアップのタイプを指定します。オフラインバックアップの場合、このコマンドを実行する前に NNMi を停止するか、-force オプションを指定してください。 値が指定されなければ、オンラインバックアップが行われます。
-scope (config|topology|events|all)	バックアップするデータを指定します。 • ファイルシステムデータの場合、領域はオンラインとオフラインの両バックアップに適用されます。 • データベーステーブルの場合、領域はオンラインバックアップのみに適用されます。オフラインバックアップの場合、領域にかかわらず組み込みデータベース全体がバックアップされます。 値が指定されなければ、完全バックアップが行われます。

表 10　**nnmbackup.ovpl** コマンドラインのオプション

オプション	説明
-force	NNMi を、強制的にバックアップのタイプに適した状態にします。 オンラインバックアップの場合、このコマンドは nmsdbmgr サービスを、まだ実行していない場合に起動します。 オフラインバックアップの場合、このコマンドはすべての NNMi サービスを停止します。 このオプションが指定されてなく NNMi が指定されたバックアップタイプにとって正しい状態でなければ、スクリプトは失敗します。
-archive	アーカイブファイル (tar ファイル) をターゲットディレクトリに作成します。
-target < *ディレクトリ* >	バックアップファイルを入れるディレクトリを指定します。

設定領域

設定領域 (`-scope config`) には、NNMi 設定情報のみが保管されるファイルおよびデータベーステーブルが含まれます。設定領域は、大まかには NNMi コンソールの「**設定**」ワークスペース内の情報と一致します。

表 11 は、設定領域用にバックアップされるファイルとディレクトリを記載しています。

表 11　設定ファイルとディレクトリ

ディレクトリまたはファイル名	説明
$NnmInstallDir/conf（Windows のみ）	設定情報
$NnmDataDir/conf	他の HP 製品が共有する設定
$NnmDataDir/shared/nnm/conf	NNMi 内の共有設定データ
$NnmDataDir/shared/nnm/lrf	NNMi 内の共有設定データ — コンポーネント登録ファイル
$NnmDataDir/shared/nnm/databases/Postgres	NNMi 組み込みデータベースのファイルシステム記憶域 **注 :** このディレクトリは、オフラインバックアップの場合のみバックアップされます。
$NnmDataDir/NNMVersionInfo	NNMi バージョン情報ファイル
$NnmInstallDir/misc/nnm	その他の設定データ
$NnmInstallDir/newconfig	インストール設定ステージング領域
$NnmInstallDir/nonOV/jboss/nms/server/nms/conf	jboss の設定

表 11　設定ファイルとディレクトリ（続く）

ディレクトリまたはファイル名	説明
$NnmInstallDir/nonOV/jboss/nms/server/nms/deploy	jboss 配備ディレクトリ
$NnmInstallDir/snmp_mibs（Windows のみ） $NnmDataDir/share/snmp_mibs（UNIX のみ）	SNMP MIB の情報
$NnmDataDir/HPOvLIC/LicFile.txt	ライセンス情報

トポロジ領域

トポロジ領域 (`-scope topology`) には、NNMi ネットワークトポロジ情報が保管されるファイルおよびデータベーステーブルが含まれます。トポロジ領域は、大まかには NNMi コンソールの**「インベントリ」**ワークスペース内の情報と一致します。

ネットワークトポロジが依存している設定はそのトポロジの検出に使用されているため、トポロジ領域には設定領域が含まれます。

イベント領域

イベント領域 (`-scope event`) には、NNMi ネットワークインシデントおよびイベント情報が保管されるファイルおよびデータベーステーブルが含まれます。イベント領域は、大まかには NNMi コンソールの**「インシデントの参照」**ワークスペース内の情報と一致します。

イベントはこれらのイベントに関連したネットワークトポロジに依存しているため、イベント領域には設定領域とトポロジ領域が含まれます。

完全バックアップ

完全バックアップ (`-scope all`) には、NNMi のすべての重要ファイルおよびデータが含まれます。設定領域、トポロジ領域、およびイベント領域が結合されています。

NNMi データのリストア

NNMi のリストアスクリプトは、指定されたソースディレクトリのデータを、ファイルシステムおよび組み込みデータベース内の適切な場所にコピーします。部分バックアップから、または完全バックアップからデータをリストアできます。

- NNMi のリストアスクリプトが機能するのは、NNMi バックアップスクリプトで収集された情報のみです。保存したバックアップデータを NNMi 管理サーバーからアクセス可能なディレクトリにコピーして、それからリストアスクリプトを実行してそのデータを目的の場所にコピーしてください。
- NNMi のあるバージョンから次のバージョンへデータベースのスキーマが変わる恐れがあるため、データバックアップを NNMi の異なるバージョン間で共有することはできません。

リストアスクリプトを実行する前に、すべての NNMi サービスを停止してください。データをオンラインバックアップからリストアしている場合は、`-force` オプションを使用してください。

リストア中に、ターゲットデータベースは完全にクリーンアップされて、バックアップの内容に置き換わります。このクリーンアップは、設定情報をあるシステムから別のシステムへ（たとえばハードウェアが新しくなったため）移動するような場合に便利です。

指定されたソースが tar ファイルの場合、リストアコマンドによって tar ファイルが抽出されて現在の作業ディレクトリの一時フォルダへ送られます。この場合、現在の作業ディレクトリに十分な記憶領域があるため一時フォルダを使用できることを確認するか、リストアコマンドを実行する前にアーカイブを抽出してください。

バックアップを別の NNMi 管理サーバーからリストアしている場合は、NNMi コンソールログインページからライセンス情報を確認して、ライセンス情報が新しいシステムに合っているかどうかを確認してください。リストアしたライセンスが新しい NNMi 管理サーバーに有効でない場合は、新しいライセンスを入手してから新しい NNMi 管理サーバーに適用してください。詳細については、ライセンスのマニュアルを参照してください。

リストアスクリプトの構文を以下に示します。表 12 では、スクリプトのコマンドラインのオプションを説明しています。

```
nnmrestore.ovpl [-force] -source <ディレクトリ>
```

表 12　nnmrestore.ovpl コマンドラインのオプション

オプション	説明
-force	NNMi を、強制的にリストアのタイプに適した状態にします。 すべてのリストアで、このコマンドはすべての NNMi サービスを停止します。 オンラインバックアップのリストアの場合、このコマンドは nmsdbmgr サービスをリストアプロセスの適切な時点で起動します。
-source < *ディレクトリ* >	バックアップファイルを入れるディレクトリを指定します。

組み込みデータベースのみをバックアップおよびリストアする

NNMi には組み込みデータベースのみをバックアップおよびリストアするスクリプトがあります。これは、NNMi の設定を試す際にスナップショットを作成する場合に便利です。組み込みデータベースのバックアップおよびリストアスクリプトでは、オンラインバックアップのみを行います。最低でも nmsdbmgr サービスが稼働している必要があります。

ベストプラクティス

組み込みデータベースにデータをリストアする前に、組み込みデータベースリセットスクリプト (nnmresetembdb.ovpl) を実行してください。このスクリプトによりデータベースにエラーが含まれないようになるため、データベース制約違反の発生する可能性がなくなります。組み込みデータベースリセットスクリプトの実行については、nnmresetembdb.ovpl リファレンスページか UNIX のマンページを参照してください。

組み込みデータベースのバックアップスクリプトの構文を以下に示します。表 13 では、スクリプトのコマンドラインのオプションを説明しています。

```
nnmbackupembdb.ovpl [-force] -target <ファイル>
```

表 13　nnmbackupembdb.ovpl コマンドラインのオプション

オプション	説明
-force	nmsdbmgr サービスを、まだ実行していない場合に起動します。 このオプションを指定しない状態で nmsdbmgr サービスを実行していないと、スクリプトは失敗します。
-target < *ファイル*>	バックアップファイルの名前を指定します。

組み込みデータベースのリストアスクリプトの構文を以下に示します。表 14 では、スクリプトのコマンドラインのオプションを説明しています。

```
nnmrestoreembdb.ovpl [-force] -source <ファイル>
```

表 14　nnmrestoreembdb.ovpl コマンドラインのオプション

オプション	説明
-force	nmsdbmgr サービスを、まだ実行していない場合に起動します。 このオプションを指定しない状態で nmsdbmgr サービスを実行していないと、スクリプトは失敗します。
-source < *ファイル*>	バックアップを入れるファイルの名前を指定します。

NNMi 管理サーバーの変更

他のシステムで HP Network Node Manager i Software 設定を複製できます。たとえば、テスト環境から生産環境に移動できますし、NNMi 管理サーバーのハードウェアを変更できます。

NNMi 設定に影響を及ぼさずに、NNMi 管理サーバーの IP アドレスを変更できます。

この章では、以下の内容を記載しています。

- NNMi 設定移動の準備のベストプラクティス
- NNMi 設定の移動
- NNMi 管理サーバーの IP アドレスの変更

NNMi 設定移動の準備のベストプラクティス

次のベストプラクティスは、NNMi 設定の異なるシステムへの移動に適用されます。

- ノードグループ設定が管理対象ノードの識別にホスト名を使っている場合、製品およびテストの NNMi 管理サーバーは同じ DNS サーバーを使う必要があります。製造システムとテストシステムが異なる DNS サーバーを使っている場合、管理対象ノードの解決済みの名前を変更すると、2 つの NNMi 管理サーバーの間でポーリング設定が異なる結果になることがあります。
- 設定エクスポートを 1 人の作成者に制限できます。自分のグループまたは会社に一意の新しい作成者値を作成します。次のアイテムを作成または変更するときに、この作成者の値を指定します。
  - デバイスプロファイル
  - インシデントの設定
  - URL アクション
- スマートプラグイン (iSPIs) をインストールする場合は、統合とプラグインセクションの該当する章を参照してください。

NNMi 設定の移動

nnmconfigexport.ovpl コマンドを使用して、NNMi 設定を XML ファイルに出力します。次に、nnmconfigimport.ovpl コマンドを使って、XML ファイルから新しいシステムの NNMi にこの設定をインポートします。

これらコマンドの詳細は、該当するリファレンスページ、または UNIX のマンページを参照してください。

NNMi 設定のみを移動できます。HP は、ある NNMi 管理サーバーから異なる NNMi 管理サーバーへのトポロジまたはインシデントデータの移動をサポートしません。また、HP は、NNM iSPI for Performance 用に収集されたパフォーマンスデータのような iSPI データの移動をサポートしません。

NNMi 管理サーバーの IP アドレスの変更

NNMi 管理サーバーの IP アドレスを変更する必要がある場合は、以下の手順に従います。

1. **http://www.webware.hp.com** にアクセスします。
2. **[ライセンスの管理]** をクリックします。
3. ログインしてから、移動プロセスを完了する手順に従って新しいライセンスキーを取得します。
4. NNMi 管理サーバーを新しい IP アドレスで設定します。
5. NNMi 管理サーバーの新しい IP アドレスを認識するように DNS サーバーを設定します。
6. NNMi 管理サーバーを再起動します。
7. コマンドプロンプトで、以下のコマンドを入力します。

   ```
   nnmlicense.ovpl NNM -g
   ```

8. **[Autopass : ライセンス管理]** ダイアログボックスで、**[ライセンスキーの削除]** をクリックします。
9. 削除するライセンスキーを選択します。
10. **[ライセンスを恒久的に削除]** を選択します。
11. **[削除]** をクリックしてから、ダイアログボックスを閉じます。
12. 手順 3 で取得した新しいライセンスキーを license.txt という名前のテキストファイルにコピーします。
13. 以下のコマンドを実行します。

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

NNM 6.x/7.x からの移行

この項では以下の章について説明します。

- 製品の比較
- NNM 6.x/7.x からの移行
- NNM 6.x または NNM 7.x と NNMi 8.10 との統合

製品の比較

この章では、HP Network Node Manager (NNM) 6.x/7.x と HP Network Node Manager i Software の間の重要な違いについて説明します。NNM の前バージョンを使用していた方は、この章を参照しながら NNMi の計画を立てたり設定してください。NNMi を初めてお使いになる方は、この章を読む必要はありません。

この章では、以下の内容を記載しています。

- ネットワーク検出
- ステータスモニタリング
- イベントモニタリングのカスタマイズ

ネットワーク検出

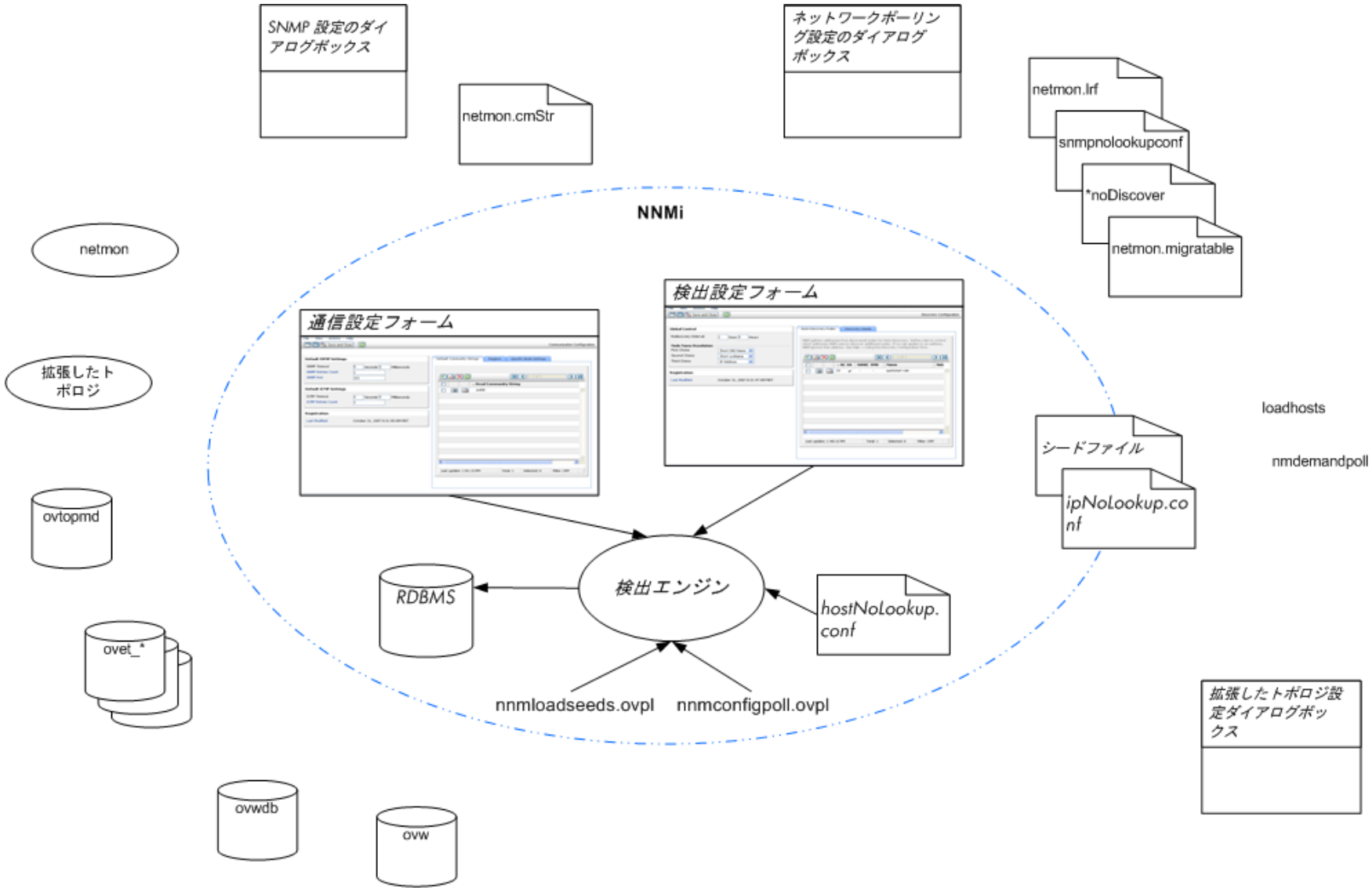
検出は、データベースに追加されているネットワークの要素（デバイス、ノード、およびこれらのコンポーネント）に対して行われます。NNMi では、「インベントリ検出」とは新しいノードを探し出すことであり、「レイヤ 2 検出」とはかつて Extended Topology 検出によって行われていた接続性モデリングを指します。

NNM 6.x/7.x のデフォルトでは、NNM は起動すると、自身のループバックアドレスをシードとして使用し、直接接続しているネットワークの自動検出を（自身の IP アドレスおよびサブネットマスクに基づいて）開始していました。NNMi では、最初から管理者制御が可能です。NNMi 自動検出では、検出を行う前に、検出領域を IP アドレス範囲に基づいて定義し、少なくとも 1 つのシードデバイス（通常はルーター）を指定します。

図 8 の中央に、NNMi での検出を設定するために使用するツール、ファイルおよびコマンドを示しています。周囲には、NNM 6.x/7.x の類似品目を示しています。

Extended Topology 情報は、Extended Topology アドオンを備えた NNM 6.x または NNM 7.x Advanced Edition のみに適用されます。

図 8　検出の設定要素


SNMP 設定のダイアログボックス
netmon.cmStr
ネットワークポーリング設定のダイアログボックス
netmon.lrf
snmpnolookupconf
*noDiscover
netmon.migratable
NNMi
netmon
通信設定フォーム
検出設定フォーム
拡張したトポロジ
loadhosts
シードファイル
nmdemandpoll
ipNoLookup.conf
ovtopmd
RDBMS
検出エンジン
hostNoLookup.conf
ovet_*
nnmloadseeds.ovpl
nnmconfigpoll.ovpl
拡張したトポロジ設定ダイアログボックス
ovwdb
ovw

検出の重要概念

ここでは、NNM 6.x/7.x から NNMi への変更の主な領域を簡単に説明します。NNMi 検出についての詳細は、NNMi ヘルプの *「ネットワークの検出」*を参照してください。

- NNMi は、すべての情報を 1 つのリレーショナルデータベース内に保管します。
- NNMi は、設定が容易な 1 つの統合検出エンジンを使用します。
- NNMi のスパイラル検出プロセスにより、ネットワークに変化が生じた際のトポロジ情報の継続的な更新が可能になります。トポロジの変化（インベントリとレイヤ 2 の両方）を、定期的な再検出間隔より頻繁に検出できます。
- NNMi では、すべての検出対象ノードは、管理モード（管理対象、管理除外、またはサービス停止）にかかわらず、ライセンス限度に対してカウントされます。ライセンス限度を超えるノードは検出できません。
- 自動検出は、NNMi と NNM 6.x/7.x では同じ意味を持っていますが、設定アプローチは異なります。
  - NNMi では、自動検出境界を定義し、少なくとも 1 つの IP アドレスシードを指定してから、検出を実行させます。
  - NNMi 自動検出では、管理が容易な拡大式モデルを使用します。NNMi 自動検出では、指定された境界内のすべてのルーター、スイッチおよびサブネットを見つけ出して管理します。NNMi で検出して管理する追加デバイスタイプを指定します。

    ➤ デフォルトでは、SNMP 以外のノードは NNMi で検出*されません*。

- シード検出は、NNMi と NNM 6.x/7.x では同じ意味を持っていますが、設定アプローチは異なります。
  - NNMi では、検出シードをユーザーインタフェースで指定します。
  - NNM 6.x/7.x シードファイルを NNMi で、変更することなく使用できます。
  - NNMi `nnmloadseeds.ovpl` コマンドは、NNM 6.x/7.x `loadhosts` コマンドに代わるコマンドです。
- NNMi 設定ポーリング (`nnmconfigpoll.ovpl`) は、デバイス設定情報を決定するための NNM 6.x/7.x 需要ポーリング (`nmdemandpoll`) に代わるものです。

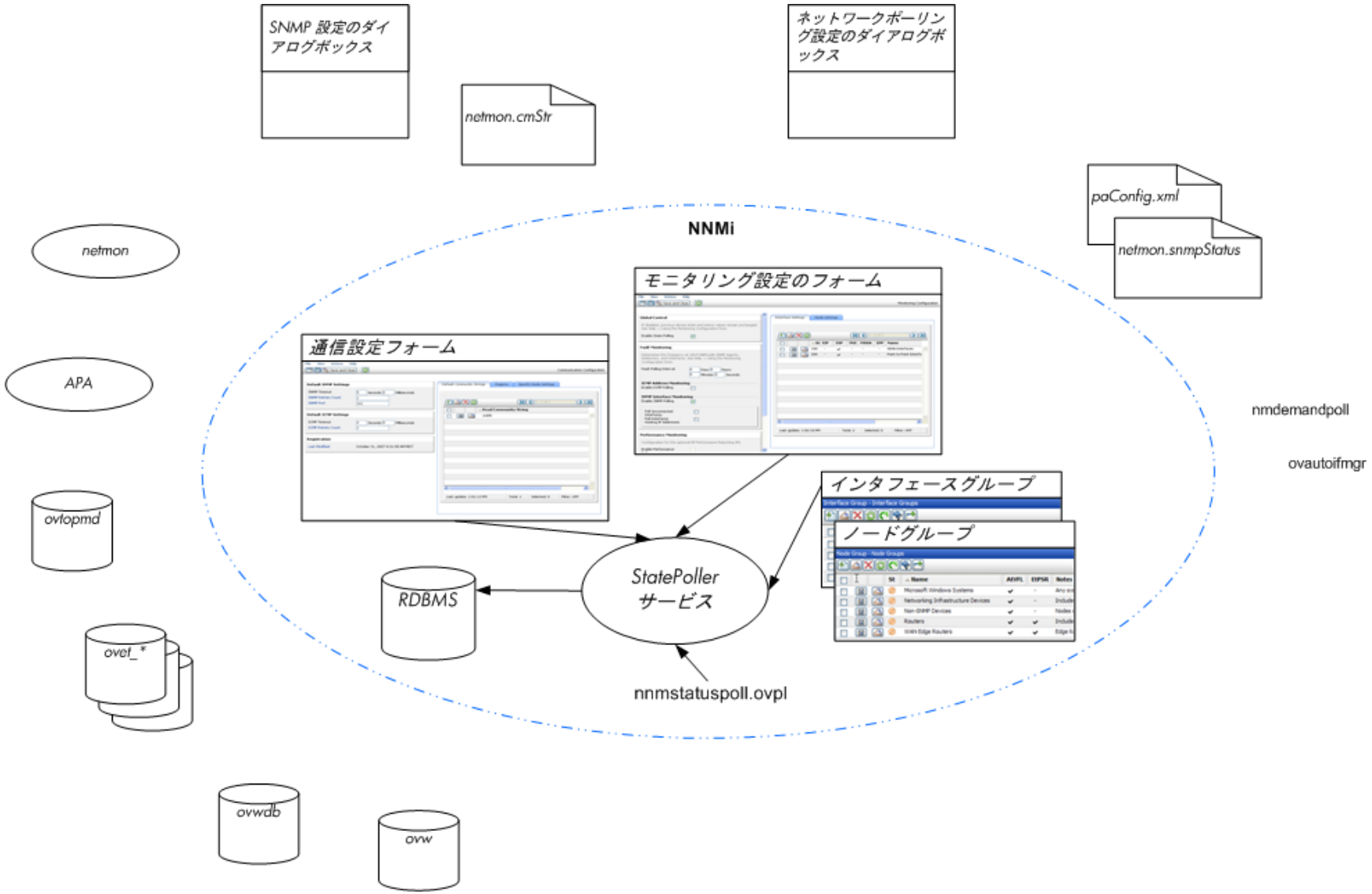
ステータスモニタリング

ステータスモニタリングを行うことにより、故障が起こり得るデバイスやコンポーネントの点から見て最新のネットワーク可視化が実現します。ある部品がポーリングに失敗すると、NNMi は原因を調査して、根本原因アラームをインシデントブラウザに送出します。

図 9 の中央に、NNMi でのステータスモニタリングを設定するために使用するツール、ファイルおよびコマンドを示しています。周囲には、NNM 6.x/7.x の類似品目を示しています。

APA 情報は、NNM 7.x Advanced Edition のみに適用されます。

図 9　モニタリングの設定要素



ステータスモニタリングの重要概念

ここでは、NNM 6.x/7.x から NNMi への変更の主な領域を簡単に説明します。NNMi ステータスモニタリングに関する詳細は、NNMi ヘルプの*「ネットワークの稼働状態のモニタリング」*を参照してください。

- 設定は、これでユーザーインタフェースを通じて完了しています。
- NNMi ノードグループおよびインタフェースグループは、トポロジフィルタに代わるものです。
  - グループは、定義済みの属性のセットでのみフィルタリングできます。
  - グループをブール演算子と連結できません。
  - ノードグループは、sysObjectId ワイルドカードを使用する代わりにデバイスフィルタを使用します。
  - インタフェースグループを、収容側ノードおよびインタフェースタイプのグループに基づいて制限できます。
- ICMP ポーリングは、デフォルトでは無効です。
- 広範な制御機能により、不要なインタフェースの除外が容易です。
- モニタリング設定は、(1) インタフェース設定、(2) ノードの設定、(3) デフォルトのように、固有性の高いものから一般性の高いものへの方向に適合しています。
- モニタリングの動作をシステム全体で変更するには、すべての設定をすべてのレベルで変更します。
- NNMi のステータスポーリング（**[アクション] > [ステータスポーリング]** または `nnmstatuspoll.ovpl`）は、デバイスのステータスを判定するための NNM 6.x/7.x の需要ポーリング (`nmdemandpoll`) に代わるものです。
- デフォルトでは、NNMi がポーリングするインタフェースは、レイヤ 2 接続を通じて別の既知のインタフェースに接続しているインタフェースのみです。接続していないインタフェースのポーリングと IP アドレスをホストしているインタフェースのポーリングを有効にできます。

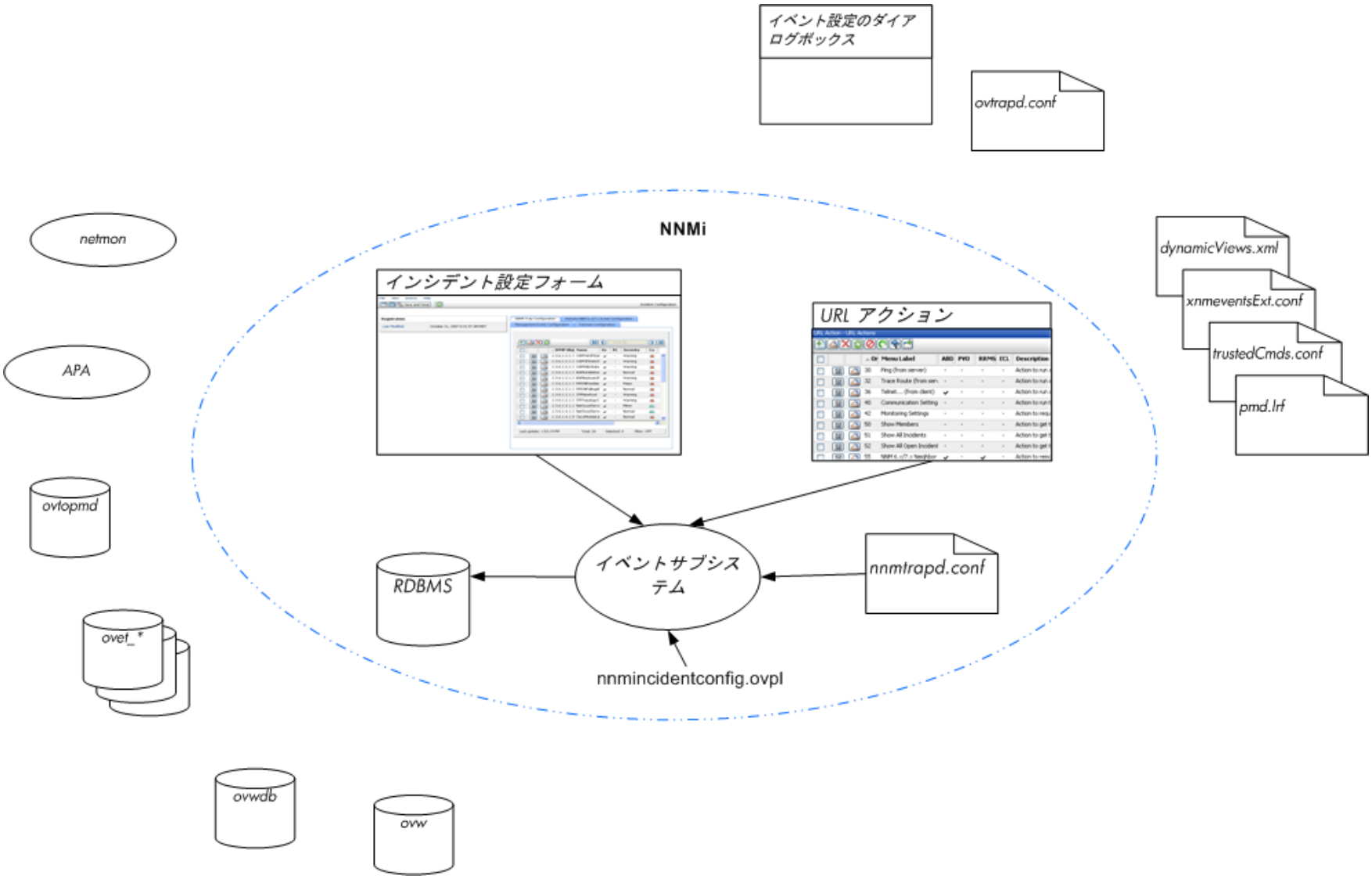
イベントモニタリングのカスタマイズ

NNMi にはインシデントビューという 1 つの中心となる場所があり、そこで管理イベント、SNMP トラップ、および NNM 6.x/7.x 転送イベントをチームが見ることができます。どの SNMP トラップや NNM 6.x/7.x イベントがインシデントとして表示するほど重要であると見なすかをコントロールしてください。

図 10 の中央に、NNMi でのイベントモニタリングを設定するために使用するツール、ファイルおよびコマンドを示しています。周囲には、NNM 6.x/7.x の類似品目を示しています。

APA 情報は、NNMi 7.x Advanced Edition のみに適用されます。

図 10　イベントモニタリングの設定要素

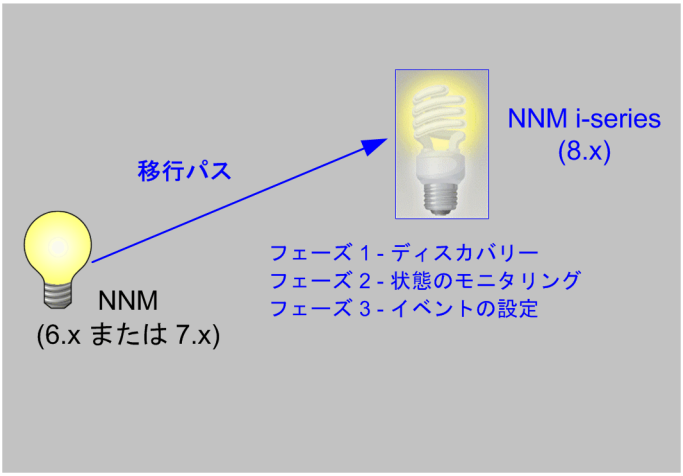


イベントモニタリングの重要概念

ここでは、NNM 6.x/7.x から NNMi への変更の主な領域を簡単に説明します。NNMi インシデントに関する詳細は、NNMi ヘルプの*「インシデントを設定する」*を参照してください。

- NNMi では、イベントサブシステムがプロセス間通信で使用されておらず、イベントのボリュームが大きく削減されています。管理者は、各々の IPC メッセージを表示するかログするかを設定する必要がなくなりました。
- NNMi は、受信するよう設定したトラップのみを受信します。設定されていないトラップは、フィルタリングされてイベントパイプラインから除かれます。
- NNMi は、受信するすべてのトラップを表示します。
- NNMi イベントサブシステムプロセスのトラップフィルタは、**[インシデントの設定]** フォームでの選択内容に基づいて、明示的に設定されます。
- NNMi `nnmincidentconfig.ovpl` コマンドは、指定された MIB ファイルのトラップ定義のみをロードします。
- NNMi は、イベントパイプラインで生じるペアワイズ、レート、および重複排除の相関を提供します。(NNMi には、イベント相関システム (ECS) は含まれません。)
- NNMi では、インシデントのライフサイクルの任意の時点で生じるアクションを設定できます。あらゆるスクリプト、実行ファイル、または Jython アクションをアクションとすることができます。
- NNMi URL アクション設定は、イベントの結果として行われるアクションを定義するための `dynamicViews.xml` および `xnmeventsExt.conf` 設定ファイルに代わるものです。

NNM 6.x/7.x からの移行



この章では、HP Network Node Manager 6.x または 7.x から HP Network Node Manager i Software への基本移行パスについて説明します。このパスは、ほとんどのユーザーのニーズに役立ちます。この章では、高度な移行のトピックまたはカスタマイズについては説明しません。これらについては、ニーズを満たすコンサルティングサービスが利用できます。

この章では、次の製品命名規約を使用します。

- **NNM** は、HP Network Node Manager の旧バージョンです（NNM の 6.x と 7.x の全リリースを含みます）。
- **NNMi** は HP Network Node Manager i Software です（NNMi と NNMi Advanced の 8.x の全リリースを含みます）。

この章では、以下のことを想定しています。

- *『NNMi インストール ガイド』の「」の指示に従って NNMi をインストールしてあります。*
- NNMi ヘルプとこのガイドの導入情報に説明してある概念、および NNMi 機能を全般的に理解していることを確認してあります。
- NNMi コンソール の使用法を理解しています。

移行ツールがリリースされるとき、そして NNMi が発展するとき、この章の情報はしばしば更新されます。

最新でダウンロード可能な NNM と NNMi のドキュメントのコピーを、次のサイトから入手してください。

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

この章では、以下の内容を記載しています。

- 移行オプション
- フェーズ 1：SNMP 情報の移行
- フェーズ 2：検出の移行
- フェーズ 3：ステータスモニタリングの移行
- フェーズ 4：イベント設定とイベント削減の移行

移行オプション

新たな始まり

数世代のソフトウェア環境、さまざまなネットワーク環境が備わっている NNM システムは多数あります。経路が決まっている世界で NNM 4.x または 5.x から開始したユーザーは、現在のネットワーク構造には実際には適合しない不必要なお荷物を進展させていることがあります。NNM システムが 2 年以上前のものである場合は、この機会を利用して新しいシステムを開始することを本気で考えてください。現在のネットワークの管理法をすみずみまで再評価すると、使用中の NNM と比較して、オーバーヘッドが大幅に減少し、操作が簡単になる可能性があります。

新しい NNMi システムの開始を選択した場合は、『*NNMi インストール ガイド*』の指示に従って NNMi をインストールしてください。次に、この『*NNMi 導入および移行ガイド*』の他の章に説明してある複雑な導入作業について考えてください。その場合は、移行についてのこの章を読む必要はありません。

フェーズのある移行

組織によっては、フェーズのある移行作業の方が、まったく新しいインストールよりうまく機能します。このような組織では、新しい NNMi システムで、既存の NNM システムを完全に再生し、置き換える必要があります。この目的に進む道は多数ありますが、HP は次のフェーズをお勧めします。

- フェーズ 1：SNMP 情報の移行

  使用中の環境の SNMP アクセス情報で NNMi を設定します。

- フェーズ 2：検出の移行

  NNM がオブジェクトを（自動）検出したのと類似した方法で、NNM が検出したオブジェクトを NNMi が検出するように設定します。

- フェーズ 3：ステータスモニタリングの移行

  使用中の環境に最も適切なステータスポーリング間隔とプロトコルを設定します。

- フェーズ 4：イベント設定とイベント削減の移行

  イベントの重要度、カテゴリ、メッセージを表示し、NNM で設定した自動処理を実行するように NNMi を設定します。重複解除、レート計算、ペアワイズのキャンセル、しきい値監視を設定する必要があることもあります。

- フェーズ 5：グラフィカルな視覚化を移行

  NNM `ovw` のカスタマイズおよびコンテナビューコンテナ（container views containers）の移行はこの時点では利用できません。

NNMi は、既存 NNM システムの管理者のマネージャとして機能できます。イベントを NNMi に転送するように NNM を設定できます。次に、統合ユーザーインタフェース、インシデント所有権、ライフサイクル状態がある NNMi コンソールを使用して、おなじみの NNM ツールに移動できます。NNM を NNMi に統合する場合の指示は、NNM 6.x または NNM 7.x と NNMi 8.10 との統合ページの 173 を参照してください。

表 15 に、移行の複雑さの範囲の両端について、移行プロセスの高レベルの概要を示します。

- 最も簡単な方法としては、NNM から環境に特有の情報をインポートし、改良したデフォルトの NNMi 設定値を NNM から受け入れます。
- 最も詳細で綿密な方法としては、NNM 設定を詳しく調べ、この設定を NNMi で複製します。

この章の残りの部分では、NNM 設定を NNMi に移行するプロセスを 1 つずつ行ってみます。左端のテキストは、特定の手順が移行プロセスにどのように当てはまるか示しています。

- **NNM** から収集したものは、NNM 管理ステーションで行う作業を示します。
- **NNMi** への複製は、NNMi 管理サーバーで行う作業を示します。
- **NNMi** での強化は、NNMi 管理サーバーで行うオプションの作業を示します。強化は、移行プロセスの間またはその後、いつでも実行できます。

適切な時点で、指定の作業完了の複雑さの範囲に応じて 1 つまたは複数のオプションが与えられます。

表 15　移行の範囲

フェーズ	最も簡単な方法	最も詳細で綿密な方法
SNMP 情報	1 現在使用中のコミュニティ文字列をすべてエクスポートします。 2 これらのコミュニティ文字列を NNMi にインポートし、どのコミュニティ文字列がどのノードと調和するか NNMi が選別するようにします。	1 現在使用中のコミュニティ文字列をすべてエクスポートします。 2 データファイルを変更し、内容を特定ノードのコミュニティ文字列として NNMi にインポートします。
検出	1 検出された全ノードのリストをエクスポートします。 2 データファイルを変更し、ファイルの内容を自動検出ルールのないシードとして NNMi にインポートします。	1 NNM と netmon がノードを検出する方法を決定します（シード、ロードホスト、フィルタ、その他のツール）。 2 シードおよび自動検出ルールを使用して、可能な限り厳密にこの方法を複製します。
ステータスモニタリング	NNMi デフォルトは、ほとんどのユーザー要件に合うように更新されます。デフォルト値を大幅に変更する必要がない可能性があるので、更新したデフォルト値で操作を開始します。	1 ノードの各グループについて、どのポーリング間隔とポーリング方針を NNM および netmon または APA が使用したか正確に調べます。 2 NNMi ノードグループとインタフェースグループを実装し、ポーリング間隔とポーリング方針を複製します。
イベント設定とイベント削減	1 NNM のデフォルト設定で開始します。 2 管理対象デバイスのカスタムトラップの定義を追加します。 3 必要に応じて、自動処理を追加します。	1 トラップとイベントの種類ごとに、何の NNM カスタマイズが行われたかを正確に調べます。 2 NNMi システム上で、一致するそれぞれのトラップとイベントの種類をカスタマイズします。

フェーズ 1：SNMP 情報の移行

管理対象デバイスとの通信を確立するために NNMi が使う SNMP コミュニティ文字列情報を移行します。

NNM 設定に、名前解決サービスで調べるべきでない IP アドレスまたはホスト名がある場合は、その情報を NNMi に移行します。

使用中のネットワークのカスタムデバイス用に NNMi デバイスプロファイルをカスタマイズします。

SNMP アクセスの設定

NNMi 検索には、設定と接続に関する個別の情報を収集するために、管理対象ノードへの SNMP アクセスが必要です。SNMP は、ノードおよびそれに含まれるオブジェクトの稼働状態にアクセスするために、ステータスモニタリングの間も使用されます。

NNM は、一致する領域用にリストされた順序でコミュニティ文字列を連続的に試してみて、役に立つ最初のコミュニティ文字列を使います。NNMi は、設定された全コミュニティ文字列を並行して試し、役に立つ最初のコミュニティ文字列を使います。役に立つ値が複数ある可能性のある最良のコミュニティ文字列を使います。

NNM から収集

NNM 管理ステーションには、使用中の環境の機器に SNMP がアクセスするための完全な設定情報があります。

1 NNM SNMP 設定をエクスポートするには、以下の操作の 1 つを実行します。
   - ユーザーインタフェースを開き、**[オプション] → [SNMP 設定]** を選択し、**[エクスポート]** をクリックします。ターゲットファイル `snmpout.txt` の名前を指定します。
   - 次のコマンドを実行します。

     `xnmsnmpconf -export > snmpout.txt`

NNM SNMP 情報の例

出力は次の例のようなものになります。

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

ターゲットファイルには、コロンで区切られた以下のフィールドがあります。

`target:community:proxy`（* プロキシでないことを示す）`:timeout`（1/10 秒単位）`:retries:poll interval`（秒単位）`:port:set-community:`

値の明確な解釈を知るには次のコマンドを使います（ただし、インポートでは使わないでください）。

`xnmsnmpconf -export- verbose`

`ovsnmp.conf` ファイルフォーマットの詳細は、*ovsnmp.conf* リファレンスページまたは UNIX のマンページを参照してください。

2 次のファイルで、設定された代わりのコミュニティ文字列を確認します。
   - *Windows*: `%OV_CONF%\netmon.cmstr`
   - *UNIX*: `$OV_CONF/netmon.cmstr`

NNMi に複写

コミュニティ文字列を NNMi に入力する方法を選択します。これらの各方法は、148 ページの手順 1 で作成した snmpout.txt ファイルの一意のコミュニティ文字列リストで開始します。

➤ [SNMP プロキシシステム] と [コミュニティ名の設定] の設定エリアは転送できません。

簡単な方法

最も簡単な方法としては、NNM コミュニティ文字列をすべて入力し、各デバイスに使う SNMP コミュニティ文字列を NNMi に分かるようにします。コミュニティ文字列検索はデフォルトで有効です。この機能を使うと移行を迅速に処理できます。

1. NOC（ネットワーク運営センター）に、NNMi の最初の検索の間、認証エラーを予測するように通知してください。NOC 職員は、その間、これらの認証エラーを差しさわりなく無視できます。
2. 以下の操作の 1 つを完了してください。
   - NNMi が使うフォーマットと一致するように snmpout.txt を変更します。次に、NNMi を使ってこれらの値をロードします。
   - snmpout.txt ファイルをサンプルとして使用し、NNMi の入力ファイルを手作業で構築します。次に、NNMi を使ってこれらの値をロードします。
   - 以下の手順で、値を NNMi コンソールに入力します。

   a. snmpout.txt ファイルの一意のコミュニティ文字列値のリストを調べます。

      — *Windows* : Excel で snmpout.txt ファイルを開きます。データ行を選択してから、コラム B でソートします。

      この例の場合は、次の 2 つの一意のコミュニティ文字列について考えます。

      パブリック
      mycommstr

      — *UNIX* : 以下のコマンドを実行します。

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

   b. NNMi コンソールで、**[設定]** ワークスペースから **[通信の設定]** を選択します。**[デフォルト]** タブに一意の値をすべて入力します。

   c. タイムアウト、再試行、およびポートを設定します。

変更した簡単な方法

コミュニティ文字列が使用された IP 領域ごとのグループコミュニティ文字列。局地的な値を NNMi コンソールにロードしてから、各デバイスに使用する SNMP コミュニティ文字列を NNMi が決定するようにします。簡単な方法よりも認証の失敗は少なくなります。

1 `snmpout.txt` ファイルで、NNM が使っている *IP 領域ごとの*一意の値のリストを調べます。
2 NNMi コンソールで、**[設定]** ワークスペースから **[通信の設定]** を選択します。IP 領域を作成してから、領域ごとにコミュニティ文字列を入力します。
3 タイムアウト、再試行、およびポートを設定します。

自動化された方法

`snmpout.txt` ファイルを `nnmcommload.ovpl` コマンドに必要なフォーマットに変換してから、各デバイスで使用中の個別のコミュニティ文字列をロードします。

1 NNMi ツールで使えるよう `snmpout.txt` ファイルを適合させるには、以下の方法の 1 つを実行します。
   - エディタを使って NNMi に適切なファイルを作成します。結果は次のようなものになります。

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

   - *UNIX のみ* : 以下のコマンドを実行します。

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

     このコマンドはファイル内の個別のノードについて機能します。範囲またはワイルドカードを手作業でトリムします。
2 以下のコマンドを実行します。

   ```
   nnmcommload.ovpl -u username -p password -file mysnmp.txt
   ```

3 NNMi コンソールで、デフォルトのコミュニティ文字列、および IP 範囲用のコミュニティ文字列を設定します。
4 NNMi コンソールで、タイムアウト、再試行、ポートをすべて設定します。

NNMi コンソール方法

NNMi コンソールで、**[設定]** ワークスペースから **[通信の設定]** を選択します。`snmpout.txt` ファイルの設定された値を複製します。

NNMi での強化

次の情報を使って、NNMi の通信アクセス設定を強化します。

- ホスト名ワイルドカード（IP 範囲より環境によく適合する場合）
- グローバルデフォルト、IP 範囲、および特定のノードによる ICMP タイムアウトと再試行
- ネットワークの特定のエリアへの SNMP または ICMP のアクセスを有効化または無効化
- 特定のノードについて優先される管理アドレス

NNM は、管理アドレスを選択するとき、最も小さいループバックアドレスを選択します。NNMi 8.10 は最も大きいループバックアドレスも選択します。NNMi 8.0x は最も大きいループバックアドレスを選択します。

名前解決の制限

DNS（または他の名前解決）サービスの制限が分かっている場合は、NNM と NNMi にこれらのデバイスのルックアップを避けるよう指示できます。この作業がシステムに該当しない場合は、SNMP アクセスの設定ページの 148 に進んでください。

NNM から収集

1 次のファイルを確認し、NNM がアドレスからのホスト名解決を除外する**アドレス**を調べます。
   - *Windows*: `%OV_CONF%\ipNoLookup.conf`
   - *UNIX*: `$OV_CONF/ipNoLookup.conf`
2 次のコマンドを確認し、NNM が名前からのアドレス解決を除外する**ホスト名**を調べます。

```
snmpnolookupconf dumpCache > snmpnolookup.out
```

NNMi に複写

3 アドレスを手順 1 から次のファイルに追加します。
   - *Windows*: `%NnmDataDir%\conf\ipNoLookup.conf`
   - *UNIX*: `/$NnmDataDir/shared/nnm/conf/ipNoLookup.conf`
4 ホスト名を手順 2 から次のファイルに追加します。
   - *Windows*: `%NnmDataDir%\conf\hostNoLookup.conf`
   - *UNIX*: `$NnmDataDir/shared/nnm/conf/hostNoLookup.conf`

これらの設定ファイルのフォーマットについては、*ipNoLookup.conf* と *hostNoLookup.conf* リファレンスページ、または UNIX のマンページを参照してください。

NNMi での強化

NNMi は検出の間のみルックアップを実行します。NNM 非ルックアップ設定を NNMi に複製すると、スパイラル検出操作が自動的に強化されます。

5 NNMi では、表示された名前ラベルとして、DNS ホスト名、IP アドレス、または MIB II `sysName` の使用を選択できます。これを行うには、以下の手順に従います。
   a NNMi コンソールで、**[設定]** ワークスペースを開きます。
   b **[検出の設定]** を選択します。
   c **[ノード名解決]** エリアでノード名優先を設定します。

デバイスプロファイルのカスタマイズ

NNM が SNMP クエリーからデバイスに直接に収集する設定情報もあります。デバイスの**システムオブジェクト ID**(`sysObjectID`) から*導出*される情報もあります。`sysObjectID` から NNMi 内の属性へのマップは**デバイスプロファイル**経由で行われます。デバイスプロファイルを使うと、モニタリング用にノードをグループにまとめ、表示用にノードをフィルタし、検出メンテナンス用にノードをカテゴリにまとめられます。

以下の設定エリアは転送できません。

- カスタム記号
- カスタムデータベースフィールドとデフォルト値

NNM から収集

1 NNM のお使いのバージョン用に OID ファイルのカスタマイズを決定します。

- NNM 6.4 以前はファイル `oid_to_sym`、`oid_to_type`、`HPoid2type` を使ってシステムの `sysObjectID` をデータベース属性にマップし、記号を表示していました。
- NNM 7.x は、`oid_to_sym` ファイルを `oid_to_sym_reg` ディレクトリ構造で置き換えます。

NNMi に複写

NNMi には既知のシステムオブジェクト ID について事前設定された多数のデバイスプロファイルが梱包されているので、必要なデバイスプロファイルをすぐに利用できる可能性があります。最も簡単な方法では、検出プロセスを開始し、結果を確認し、必要な場合のみ変更を行います。

2 NNMi コンソールでは、**[設定]** ワークスペースから **[デバイスプロファイル設定]** を選択します。カスタマイズした値ごとに `sysObjectID` でエントリを見つけます。

3 必要に応じてデバイスプロファイル設定を更新します。

- NNMi で利用できるエントリについては、設定された値が NNM 属性と一致することを確認します。
- NNMi 内のエントリについては、`sysObjectID` 用に新しいデバイスプロファイルを作成します。将来のリリース用に ID を追加するよう HP に通知する拡張リクエストを送信します。

ベストプラクティス

4 最初の検出の後、プロファイルごとにノードインベントリをソートして、**[デバイスプロファイルなし]** ノードを見つけます。

[デバイスプロファイルなし] という種類のプロファイルには、`sysObjectID` がまだ NNMi で設定されていません。NNMi は、**[デバイスプロファイルなし]** のノードにデフォルトのモニタリング設定を使用しますが、これらのノードはフィルタが困難です。

新しいデバイスプロファイルを構築すると、NNMi データベース内のすべてのデバイスプロファイルに `sysObjectID` が設定されます。

フェーズ 2：検出の移行

検出スケジュールと設定を移行します。NNMi スパイラル検出は、1 つまたは複数の検出シードを保存するとただちに開始します。

ネットワーク環境向けの適切なコミュニティ文字列を使用するよう NNMi を設定してから検出を開始します。

最初の検出の後、NNM で手動で設定したデバイス間の接続を移行します。

検出のスケジュール

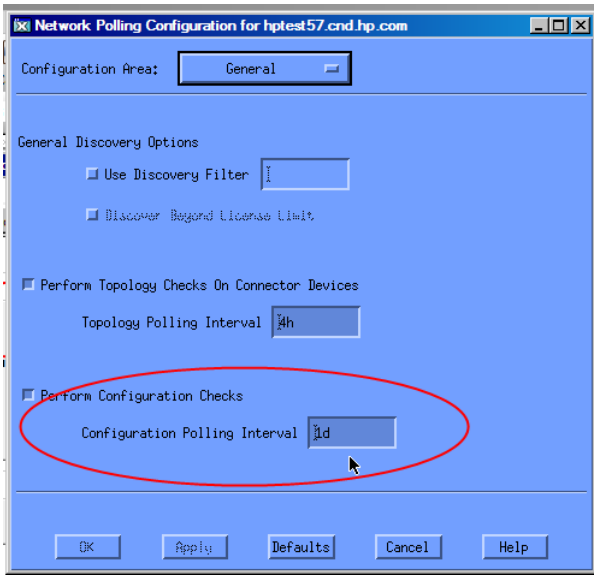
NNM 検出プロセスは独立に実行できます。検出を NNMi に移行するには、NNM がノードを検出する**間隔**を転送するだけで十分です。

以下のスケジュール設定エリアは NNMi では使用されなくなっており、転送できません。

- トポロジはコネクタデバイスをチェックします。NNMi が変更の可能性を示すトリガーを見つけるたびにトポロジチェックが自動的に行われるようになりました。
- 設定チェック。設定チェックは、スケジュールされた検出の時点、または NNMi でのトリガーの時点で行われるようになりました。
- レイヤー 2（拡張トポロジ）検出動作。NNMi は、各デバイスを見つけたときにレイヤー 2 検出を実行するので、この動作を別にスケジュールする必要はありません。
- 検出ポーリング間隔の自動調整。

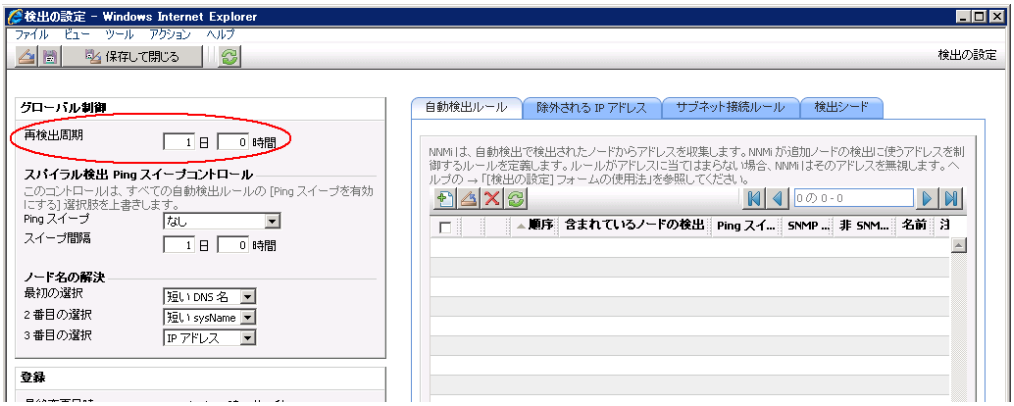
NNM から収集

1. NNM が再検出をいつ行うか決めます。
   a. ユーザーインタフェースで、**[オプション]** → **[ネットワークポーリング設定]** を選択します。
   b. **[IP ポーリング]** ページで、**[検出ポーリング間隔]** ボックスを確認します。
      — NNM が固定間隔を使う場合は、NNMi への転送の値に注意してください。
      — NNM が自動調整間隔を使う場合、NNM は最大 24 時間待機します。待機時間は 24 時間のままにしておくこともできますし、新しい値を選択することもできます。
      — 自動検出が有効になっていない場合は、**[全般]** ページの **[設定チェックを実行]** の間隔を調べ、NNMi への転送の値に注意してください。



NNMi に複写

2 NNMi コンソールで、**[設定]** ワークスペースから **[設定を検出]** を選択し、**[再検出間隔]** を手順 1 で決定した値に設定します。



NNMi での強化

他の設定更新はすべて自動的に追加されていくので、NNM よりも設定が簡単で、検出が効率的です。

自分の検出方法の選択

NNMi 検出に、次のどのモデルを使うか決定します。

- シードされた検出、自動検出ルールなし。この種類の検出は管理者によって制約されます。管理者は、必要に応じてシードを追加し、検出されるものを制御します。次の操作のみを完了します。
  - シード検出のためにシードを NNMi に追加しますページの 159
- シードと自動検出ルールに基づいた自動検出。次の両方の操作を完了してください。
  - 自動検出ルールの設定ページの 155
  - シード検出のためにシードを NNMi に追加しますページの 159

NNMi 検出方法の間の違いについては、NNMi ヘルプの「*検出方法の決定*」を参照してください。

NNM ライセンスは管理下にあるノード数に基づいて判断されます（ステータスモニタリング）。NNMi のライセンスは、検出されたノード数を基礎にして、トポロジ（モニタリングされるノードとモニタリングされないノード）に配置されます。

この違いがあるので、検出ノード数を少なくしようとする人もいるでしょうが、モニタリングされないノードをデータベースに入れると利点もあります。例：

- サービスプロバイダのアクセスルーター、およびそれへの接続を表示することができます。デバイスの管理を担当しない場合でもです。
- ステータスモニタリングアルゴリズムはデータベースに表示される接続に基づいています。データベースでリンクの他端にデバイスがないインタフェースは、デフォルトでモニタリングされません。ステータスモニタリング設定でデフォルトを上書きすることもできますし、デバイス検出を選択することもできます。どれを選択するかは、各自の環境についてどこに関心を置くかのバランスで決まります。詳細については、モニタリングされないノードへのインタフェースページの 60 を参照してください。

自動検出ルールの設定

NNMi 検出設定は、NNMi の管理対象について考える優れた機会と言えます。NNM 検出設定とフィルタの変換を行う前に、現在のネットワーク環境を考察し、NNMi トポロジに組み込むものの記述について考えてください。

直接変換を行わない場合、NNMi 検出ルールには NNM の次の 2 つのタスクセットが含まれています。つまり、検出のスコープの拡大、およびスコープで検出されるオブジェクトの制限です。

NNMi 設定の場合、検出を拡大または制限する全ルールを定義してから、検出プロセスを開始するシードを入力することが重要です。

以下のスケジュール設定エリアは NNMi では使用されなくなっており、転送できません。

- Windows から IPX 検出
- ライセンスの制限を越える検出
- レイヤー 2 オブジェクトの検出を無効化（NNMi については常に有効です）
- `netmon.interfaceNoDiscover`
- IP アドレスと `sysObjectID`（およびその派生物）以外の属性のフィルタによる検出の除外
- `bridge.noDiscover` によるレイヤー 2 検出の制限
- CDP プロトコルエリア（統合ポート、vlan など）に基づいたレイヤー 2 検出の制限
- 拡張トポロジゾーンの設定。NNMi のスパイラル検出には該当しなくなっています。

スパイラル検出の設定

NNMi には、NNMi でスパイラル検出を設定する次の 2 つの方法があります。つまり、ノードの手動でのロード（たとえば、ホストファイルから）、および自動検出ルールの使用。

ノードの手動でのロード

NNM から収集

1. NNM で、`loadhosts` コマンドの出力のあるファイルを見つけます。このファイルには、各ノードの IP アドレスとホスト名、さらに指定されている場合はサブネットマスクがリストされています。

NNM loadhosts の例

`loadhosts` コマンドのファイルの例は次のとおりです。

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

NNMi に複写

2. NNMi では、NNM `loadhosts` コマンドと同じ方法で検出シードを使用できます。これを行うには、`-f` オプションを指定して `nnmloadseeds.ovpl` コマンドを使用し、シードファイルを指定します。

シードを NNMi に設定する前に、すべてのコミュニティ文字列の設定を完了してください。

検出の出力を NNM `loadhosts` と同じにする場合は、NNMi で設定されている自動検出ルールを無効にします。自動検出ルールを無効にするには、以下の 1 つを実行します。

- **[検出の設定]** フォームからルールを削除します。
- **[自動検出ルール]** フォームで、**[含まれているノードの検出]** チェックボックスをオフにします。

NNMi のシードファイルのフォーマットでは、行ごとに IP アドレスまたはノード名（オプションでコメント付き）があります。詳細は、*nnmloadseeds.ovpl* リファレンスページまたは UNIX のマンページを参照してください。

NNMi シードファイルの例

次の例に、NNM `loadhosts` コマンドおよびホストファイルと同じ機能の NNMi シードファイルを示します。

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

ベストプラクティス

次のファイルには、拡張トポロジのデバイスリストがあります。

- *Windows*: `%OV_DB%\nnmet\hosts.nnm`
- *UNIX*: `$OV_DB/nnmet/hosts.nnm`

最初のフィールド（IP アドレス）または 2 番目のフィールド（ノード名）をコピーして NNMi のシードファイルを作成できます。

UNIX では、次のコマンドを実行してノード名のファイルを作成できます。

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

ベストプラクティス

NNMi では、管理アドレスとしてループバックアドレスが必ず優先されます。ループバックアドレスを使わない場合、NNMi では、管理アドレスとしてシードアドレスがおそらく使われれます（必ずではありません）。したがって、優先される IP アドレスのあるホストファイルをコピーするのが良いやり方です。ホスト名を使う場合は、DNS が解決されて優先される管理アドレスになることを確認します。しかし、NNMi が管理アドレスとしてこのアドレスを使うことが保証されるわけではありません。管理アドレス選択の詳細は、NNMi ヘルプの「*検出ノード名の選択*」を参照してください。

自動検出ルールの使用

NNM から収集

1. NNM に検出フィルタが使われたかどうかを調べます。NNM では、1 つの検出フィルタが検出のスコープ全体に適用されます。
   a. NNM ユーザーインタフェースを開きます。
   b. **[オプション]** → **[ネットワークポーリング設定]** を選択します。
   c. **[全般]** ページで **[フィルタ使用]** チェックボックスを確認し、オンの場合は使用中の検出ファイルに注意してください。フィルタが使用されていない場合はシード検出のためにシードを NNMi に追加しますページの 159 を続けます。
   d. 次のファイル内で検出フィルタを見つけます。
      — *Windows*: `%OV_CONF%\C\filters`
      — *UNIX*: `$OV_CONF/C/filters`

      ロジックを注意深く確認します。NNMi では、IP アドレスの範囲とシステムオブジェクト ID の範囲をフィルタできます。変換できるオブジェクトもあります。たとえば、ホスト名ワイルドカードから IP 範囲への変換、またはベンダー名からシステムオブジェクト ID 範囲への変換です。

NNM 検出フィルタの例

次の例に、NNM フィルタを示します。たとえば、ルーター、ブリッジ、Nokia_Firewalls、NetBotz、NetsNSegs です。NetBotz ファイアウォールと Nokia ファイアウォールは sysObjectID で定義されます。

```
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" ~ .1.3.6.1.4.1.94.1.21.2.10.11 ) )
||
( isNode && ( "SNMP sysObjectID" ~ .1.3.6.1.4.1.94.1.21.2.10.12 ) )
||
( 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 }
```

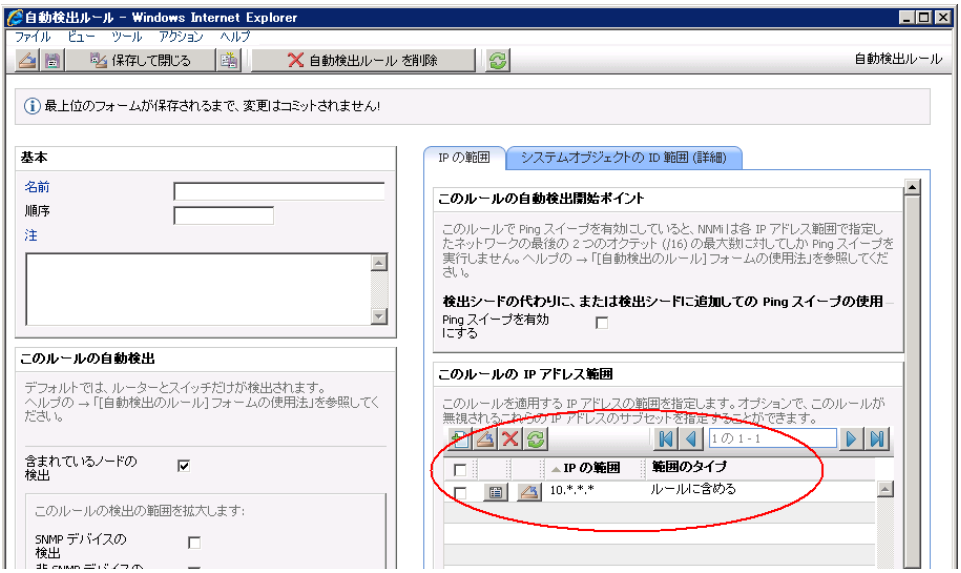
NNMi で複写

2 NNMi コンソールに検出フィルタを入力します。

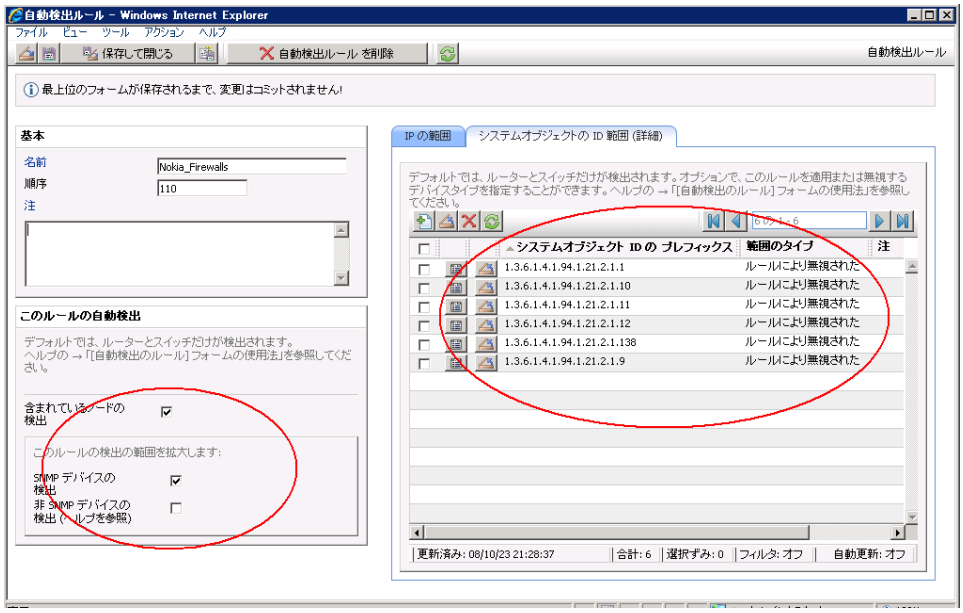
NNMi 検出フィルタエントリの例

たとえば、NNM 検出フィルタの例ページの 157 に示す NNM フィルタを NNMi に転送するには、次の 3 つの自動検出ルールを定義します。1 つのルールは Nokia ファイアウォール用、1 つのルールは NetBotz デバイス用、最後の 1 つのルールはルーターとスイッチ用です（NNM 7.x のブリッジと同じ）。NNMi では、NetsNSegs は不要です。この例の場合、検出されるネットワークの範囲は 10.*.*.* と仮定します。

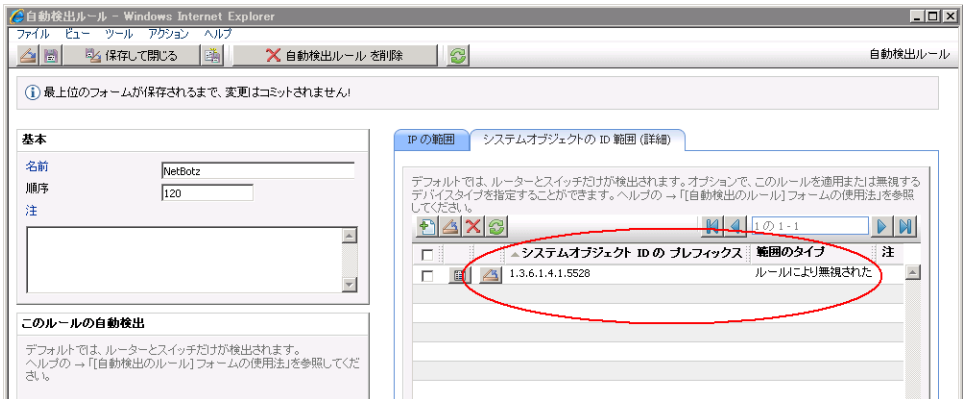
a Nokia ファイアウォールの場合、ルール名（Nokia_Firewalls）を入力してから、ネットワーク IP 範囲 10.*.*.* を入力します。



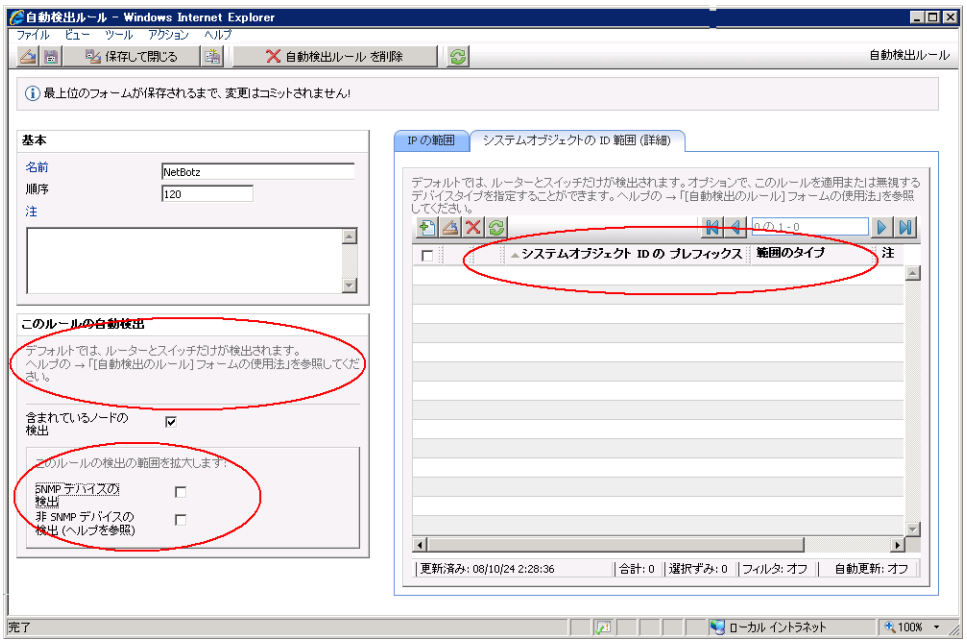
b 各 sysObjectID を入力し（先頭のピリオドは入力しません）、次に **[SNMP デバイスの検出]** チェックボックスをオンにします。（デフォルトでは、NNMi はスイッチとルーターのみを検出します。これらのデバイスはスイッチまたはルーターとマークされていないこともあるので、sysObjectIDs を指定するときに **[SNMP デバイスの検出]** チェックボックスをオンにします）。



c NetBotz ルールを入力します。このルールでは NNM `.1.3.6.1.4.1.5528.*` にワイルドカードを使います。NNMi では、アスタリスク (.*) は黙示的なので、不要です。



d 最後のルールはスイッチとルーター用です。NNMi はデフォルトでこれらのデバイスを検出するので、オブジェクト ID (OID) は指定しないでください。IP 範囲のみを指定する必要があります。



シード検出のためにシードを NNMi に追加します

NNM から収集

1 次のコマンドを実行して、NNM データベース内のデバイスの正確なリストを調べます。

```
ovtopodump > topology.out
```

NNMi で複写

2 NNM から `topology.out`（エクスポート）ファイルをコピーおよび編集します。または NNMi にインポートするために、ファイルにエントリを再入力します。新しいファイルでは、行ごとに 1 つの明確な IP アドレスまたはホスト名が記載されているはずです。NNMi がサブネットマスクを自動的に決定するので、サブネットマスクを指定する必要はありません。

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

この代わりに、NNMi コンソールを使ってノードのこのリストを追加することもできます。

3 以下のコマンドを実行します。

nnmloadseeds.ovpl f *newSeedfile*

詳細は、*nnmloadseeds.ovpl* リファレンスページまたは UNIX のマンページを参照してください。

NNMi は、これらのシードと関連付けられたデバイスの検出をただちに開始し、既存のデバイスプロファイル（およびステータスモニタリング用のノードグループなど、ノードグループ）を実装します。NNMi スパイラル検出は進行中です。検出シードの結果を知る方法については、『*NNMi インストール ガイド*』の「*検出進行状況のチェック*」を参照してください。

接続のカスタマイズ

デバイス情報が制限される特定の状況では、NNM の拡張トポロジはネットワーク内のすべての接続を正確に検出およびモデル化しない可能性があります。その結果、接続が存在することが分かっているところに接続が表示されなかったり、接続が存在しないことが分かっているところに接続が示される可能性があります。この状況を是正するためには、正しい接続を手動で作成します。

NNM から収集

1 次のファイルを確認して、NNM に接続が手動で設定されたかどうか調べます。

- *Windows*: `%OV_CONF%\nnmet\connectionEdits`
- *UNIX*: `$OV_CONF/nnmet/connectionEdits`

これらのファイルの使用法については、『*Using Extended Topology*』マニュアルまたはホワイトペーパーのディレクトリを参照してください。

NNM の接続の例

次の例に、NNM 7.x で 2 つの接続を作成する方法を示します。1 つの接続は `ifAlias` を基礎にしており、もう 1 つの接続は `ifIndex`（ボードとともに）を基礎にしています。

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

NNMi に複写

2 `nnmconnedit.ovpl` ツールを使って、NNMi で接続を編集します。ファイルフォーマットは NNM で使われているものとはまったく異なります。

a 接続テンプレートを生成するには、次のコマンドを実行します。

```
nnmconnedit.ovpl t addconn
```

詳細は、*nnmconnedit.ovpl* リファレンスページまたは UNIX のマンページを参照してください。

b テンプレートファイル（`addconn.xml`）を編集して接続を変更または追加します。新しいファイルの構文については、ファイル内のドキュメントを参照してください。

NNMi の接続の例

次の例に、手順 1 の NNM 7.x の例と NNMi で同等のものを示します。

```
<connectionedits>
        <connection>
                <operation>add</operation>
                <node>N1.mycorp.net</node>
                <interface>MyAlias</interface>
                <node>N2.mycorp.net</node>
                <interface>MyOtherAlias</interface>
        </connection>
        <connection>
                <operation>add</operation>
                <node>Y1.mycorp.net</node>
                <interface>999</interface>
                <node>Y2.mycorp.net</node>
                <interface>2</interface>
        </connection>
<connectionedits>
```

c 新しい接続情報をデータベースにロードするには、次のコマンドを実行します。

```
nnmconnedit.ovpl f addconn.xml
```

d NNMi コンソールでは、[**インベントリ**] ワークスペースから [**レイヤ 2 接続**] を選択し、結果を確認します。

フェーズ 3：ステータスモニタリングの移行

NNM 6.x では、netmon プロセスがステータスモニタリングを実行します。NNM 7.x では、netmon プロセスまたは APA がステータスモニタリングを実行します。

- netmon プロセスは、インタフェースを含むノードのようなデバイスをモデル化し、おもにモードレベルでポーリングパラメータを適用します。
- APA は、アドレス、インタフェース、集合インタフェース、ボード、およびノードをモデル化します。APA はこれらのどのレベルででもポーリングパラメータを適用できます。

NNMi では、ノード、インタフェース、またはアドレスのレベルでポーリングパラメータを適用できます。

以下の機能動作は、NNMi では使用されなくなっており、転送不能です。

- DHCP ノードの特殊な処理
- ポーリングに応答しないノードの自動削除
- ボードと集合インタフェースは、現在は NNMi でモデル化されておらず、ステータスモニタリングを備えられません。

ポーリング間隔の設定

NNM から収集

NNM netmon ポーリングプロセス

netmon プロセスが NNM 一般ポーラーである場合は、NNM ユーザーインタフェースからポーリング間隔を取得します。

NNM APA ポーリングプロセス

NNM paConfig.xml の例

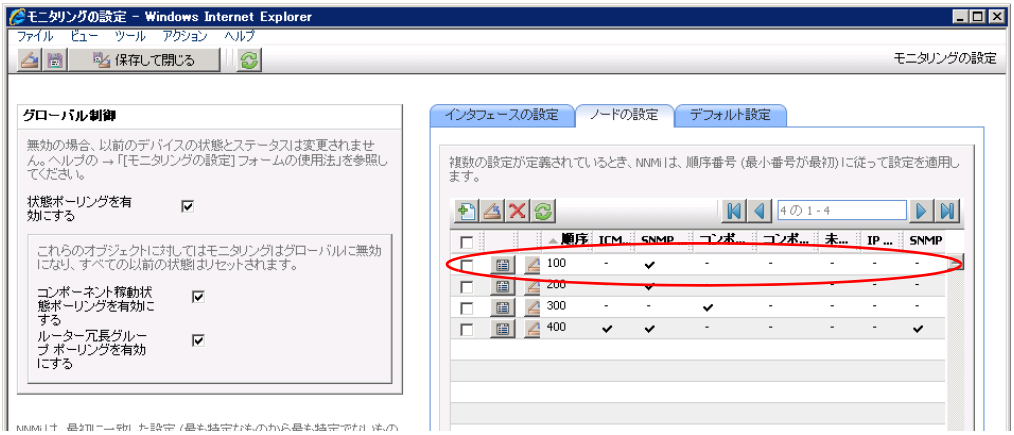
APA が NNM 一般ポーラーである場合は、paConfig.xml ファイルを見つけ、現在のポーリング間隔を知ります。例：

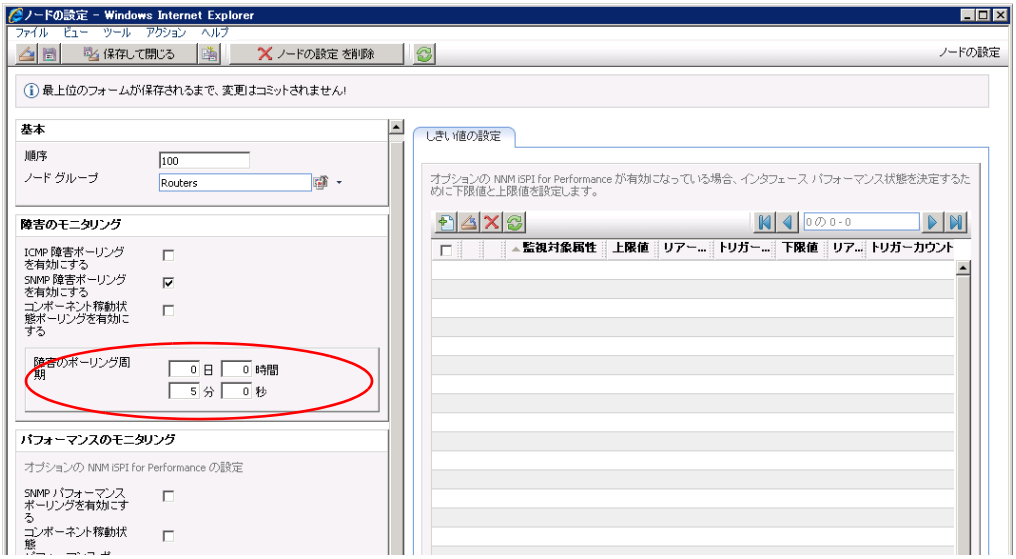
```
<classSpecification>
   <filterName>isRouter</filterName>
   <parameterList>
      <parameter>
         <name>interval</name>
         <title>Interval to Poll Device</title>
         < 説明 >
          デバイスがポーリングされる間隔
             (秒単位)。
         </description>
         <varValue>
             <varType>Integer</varType>
             <value>300</value>
         </varValue>
      </parameter>
          . . .
   </parameterList>
</classSpecification>
```

NNMi に複写　NNMi ポーリングプロセス

NNMi ステータスモニタリング設定はノードのグループまたはインタフェースのグループ（またはその両方）に基づいています。

NNMi コンソールで、**[設定]** ワークスペースから **[モニタリングの設定]** を選択します。**[ノードの設定]** タブを選択してから、グループについて **[障害ポーリング間隔]** を設定します。





ポーリングプロトコルの選択

NNM から収集　NNM netmon ポーリングプロセス

デフォルトで、netmon プロセスは ICMP を使用して各アドレスをポーリングします（各アドレスはインタフェースと同一視されます）。netmon プロセスがデバイスによっては、ICMP でなく SNMP を使うように NNM を設定することもできます（両方を使うことは決してありません）。ICMP を使っているエリアがあるかどうか調べるには、次のファイルを確認します。

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

NNM APA ポーリングプロセス

APA はポーリング用に SNMP と ICMP の組み合わせを使用します。APA では、フィルタごとにグループにまとめられたノードまたはインタフェースにポーリング方針が適用されます。フィルタは `TopoFilters.xml` ファイルで定義します。ポーリング方針は `paConfig.xml` ファイルで定義します。

NNMi で複写

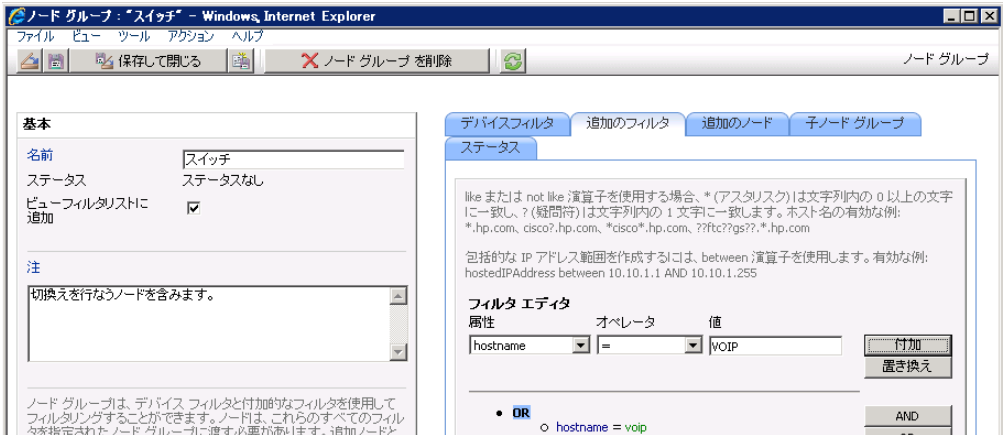
NNMi ポーリングプロセス

NNMi では、ノードとインタフェースの集合はノードグループとインタフェースグループとして定義します。ポーリング方針は **[モニタリングの設定]** フォームでノードグループとインタフェースグループに適用されます。

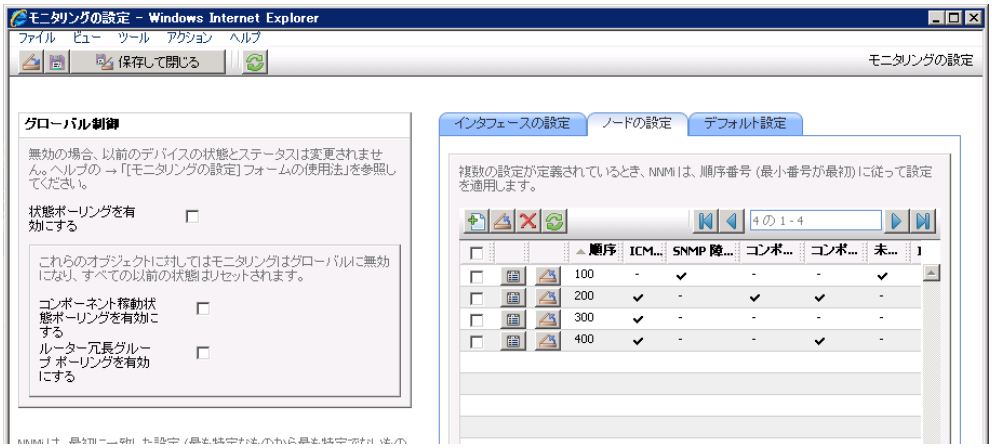
NNMi ポーリング設定の例

たとえば、(SNMP と ping を使って) VOIP ルーターの集合にポーリングを設定するには、以下の手順に従います。

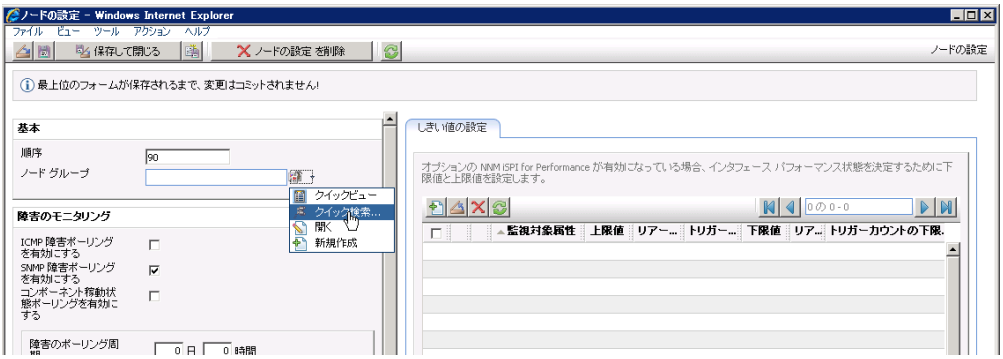
1 **[ノードグループ]** フォームを使って、VOIP ルーターを識別するノードグループを作成します。次のように、ホスト名ワイルドカードでは大文字と小文字が区別されます。このフォームを保存し、閉じます。



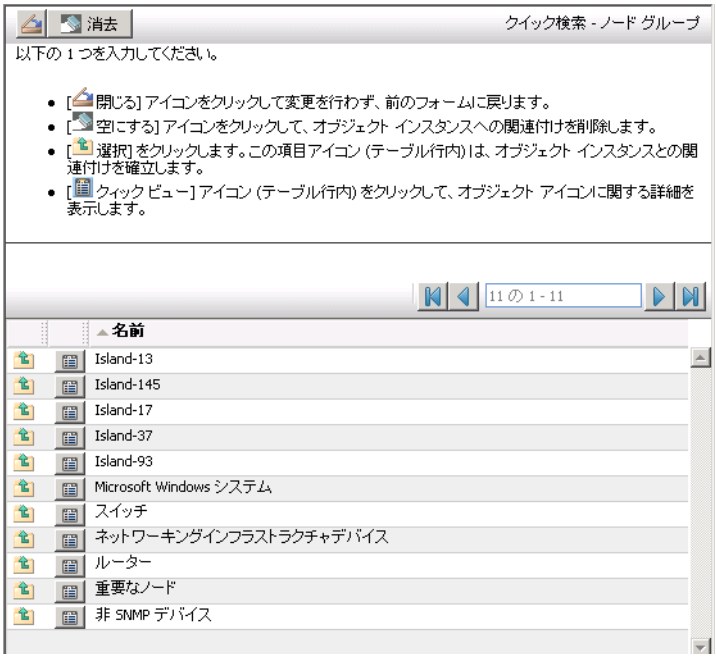
2 **[モニタリングの設定]** フォルダで、次のように、新しいノードの設定を追加します。



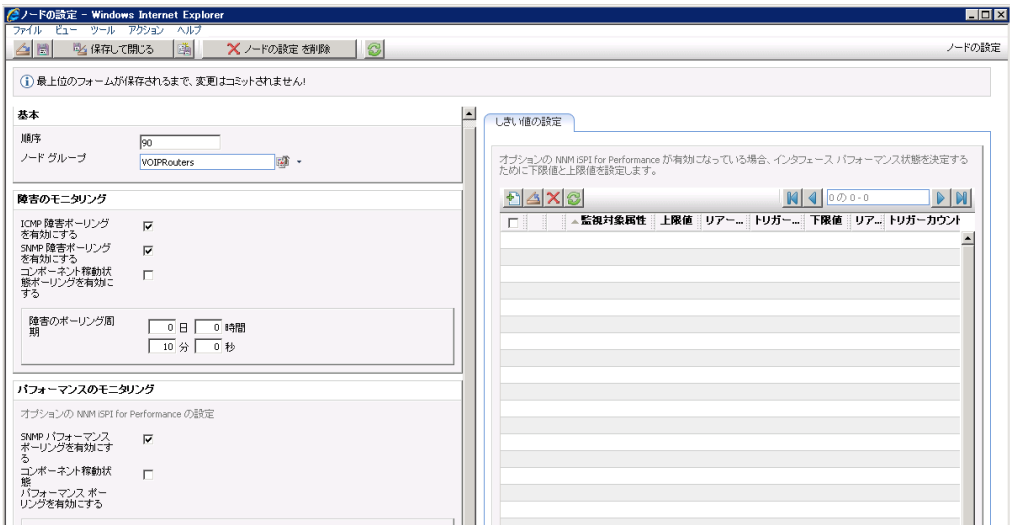
3 順序付けの値を指定してから、次のように、**[ノードグループ]** フィールドの高速検索を選択します。



4 次のように、モニタリング設定用のノードグループを選択します。



5 次のように、**[ICMP Ping アドレス障害ポーリングの有効化]** チェックボックスをオンにします。フォームを保存し、閉じます。



危険域にあるノードの設定

デフォルトで、NNMi には重要ノード用のノードグループがあります。このノードグループは、NNM の危険域にあるノードリストと同じ方法で機能します。

重要ノードが故障または到達不可能な場合、NNMi は、ノードステータスが危険域であると表示し、NodeDown インシデントを生成します。

NNM から収集

NNM netmon ポーリングプロセス

NNM がステータスモニタリングに netmon を使っている場合、NNM は危険域にあるノード用に設定されません。NNMi に新しい危険域にあるノードの設定を作成できます。

NNM APA ポーリングプロセス

次のファイルを確認し、APA についてどのノードが危険域にあると指定されたか調べます。

- *Windows*: `%OV_CONF%\nnmet\topology\filter\CriticalNodes.xml`
- *UNIX*: `$OV_CONF/nnmet/topology/filter/CriticalNodes.xml`

NNM CriticalNode.xml の例

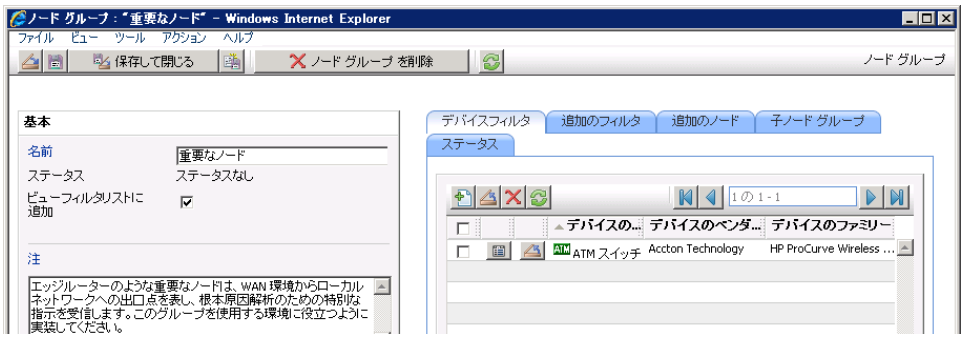
`CriticalNode.xml` ファイルは次の例に類似しています。

```
<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.mycorp.net</DNSName>
   <DNSName>router7.mycorp.net</DNSName>
   <DNSName>MPLSRtr*.mycorp.net</DNSName>
</HostIDs>
```

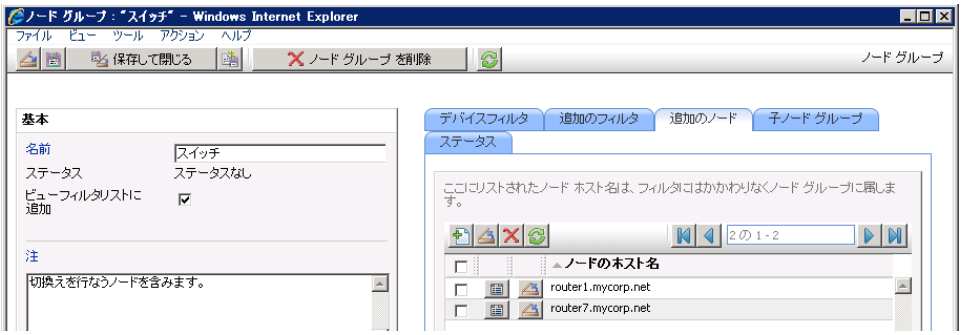
NNMi に複写

NNMi ポーリングプロセス

1. NNMi コンソールで、**[設定]** ワークスペースから **[ノードグループ]** を選択します。
2. **[重要ノード]** グループを開きます。
3. 次のように、ホスト名ワイルドカード、デバイスフィルタ、または特定のノードごとに、重要ノードをグループに追加します。
   a. ホスト名ワイルドカードを追加します。



b 特定のノードを追加します。フォームを保存し、閉じます。

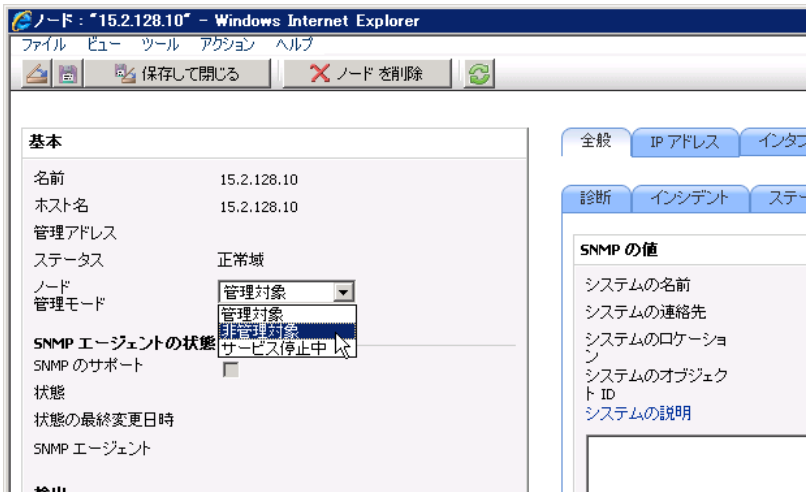


ステータスポーリングからオブジェクトを除外します

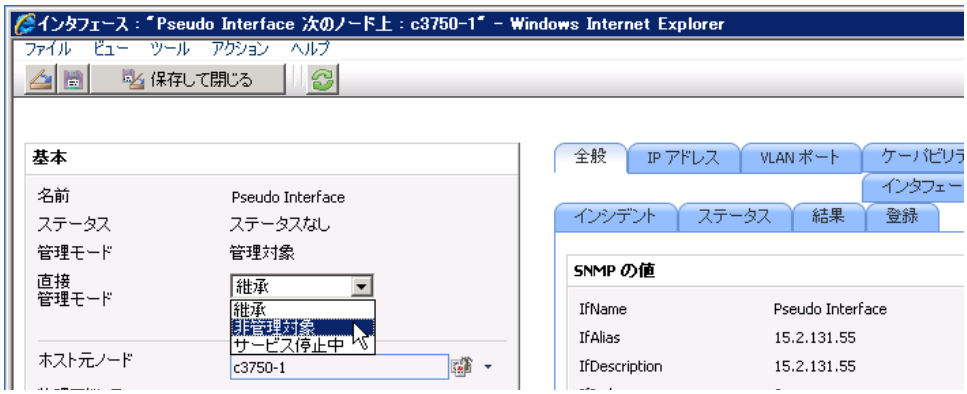
NNM では、ノードまたはインタフェースがモニタリングされるのを停止する (UNMANAGED (管理対象除外) 状態に設定する) ほとんどのアクティビティは NNM ユーザーインタフェースによって手動で完了されます。

NNMi はオブジェクトの管理除外プロセスを簡単にします。新しい製品のデフォルトを、手動で実行していたものと一致させることはできます (たとえば、アップリンクのポーリングのみ)。しかし、ノードグループとインタフェースグループを使って設定を管理すれば、設定の自動更新が簡単になります。

ノードまたはインタフェースを **Not Managed（管理対象外）** とマークする必要がある場合もあります。次のように、個別のノードの管理モードを **[ノード詳細]** フォームで設定できます。



次のように、個別のインタフェースの管理モードを **[インタフェース詳細]** フォームで設定できます。



フェーズ 4：イベント設定とイベント削減の移行

NNM は、拡張 SNMPv2 フォーマットを使って、受信イベント（管理対象デバイスからのトラップ、内部プロセス通信、転送されたイベント）の全ソースを分析します。イベントごとに、1 つのイベントオブジェクト識別子、1 つの名前、および設定パラメータがあります。

NNMi はイベントのさまざまなソースを異なるように処理します。デバイスからのトラップおよび NNM 管理ステーションから転送されたイベントは SNMPv2 フォーマットです。NNMi 内部で作成されたインシデントには名前だけがあります。イベントオブジェクト識別子はありません。さらに、NNMi 内部プロセス通信は新しい（トラップでない）メカニズムを使って、全般的なパフォーマンスを大幅に向上させています。NNMi では、認識されないイベントに関する `trapd.conf` メッセージにはフォーマットなしがありません。認識されないメッセージは破棄されるようになりました。

以下のイベント設定エリアは NNMi では使われなくなっており、転送もできません。

- イベント転送（`nnmsnmpnotify` を呼び出す自動処理でシミュレートできます）
- 構成要素相関処理の種類のいくつか：suppress（抑制）、enhance（強化）、transient（過渡的）、multisource（複数ソース）

デバイスからのトラップの表示

NNM 環境に類似した方法で、デバイスからのトラップを表示するよう NNMi を設定できます。

NNMi には、NNM に同梱されている一般的な SNMP トラップおよびベンダートラップの多くのデフォルト設定があります。これらトラップのカスタマイズによって、NNMi を更新できます。

メッセージと自動処理に利用できる変数のリストについては、NNMi ヘルプの「*Configure an Action for an Incident*」と「*Valid Parameters for Configuring Incident Actions*」を参照してください。

NNM から収集

1. NNM 設定にカスタマイズされたトラップがあるかどうか調べます。カテゴリ、重要度、表示メッセージ、または自動処理について行われたカスタマイズに注意してください。

NNMi に複写

2. ベンダー MIB ファイルを NNMi 管理サーバーにダウンロードします。
3. MIB ごとに次のコマンドを実行します。

   `nnmincidentcfg.ovpl loadTraps mibFile`

   - ある MIB に別の MIB ファイルへの依存関係がある場合は、次のコマンドを使用して依存関係を事前にロードします。

     `nnmincidentcfg.ovpl loadMib mibFile`

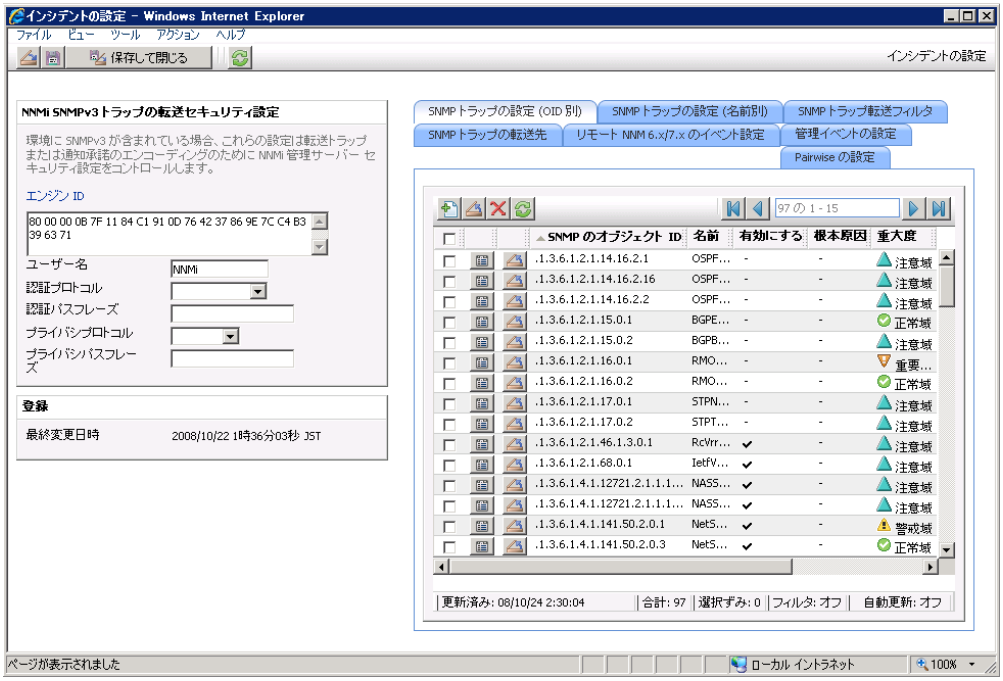
     この代わりに、**`nnmloadmib`** コマンドを使用してから、`loadTraps` オプションを指定して `nnmicidentcfg.ovpl` を再実行することもできます。
   - どの MIBs がすでにロードされているか知るには、次のコマンドを使用します。

     `nnmloadmib list`

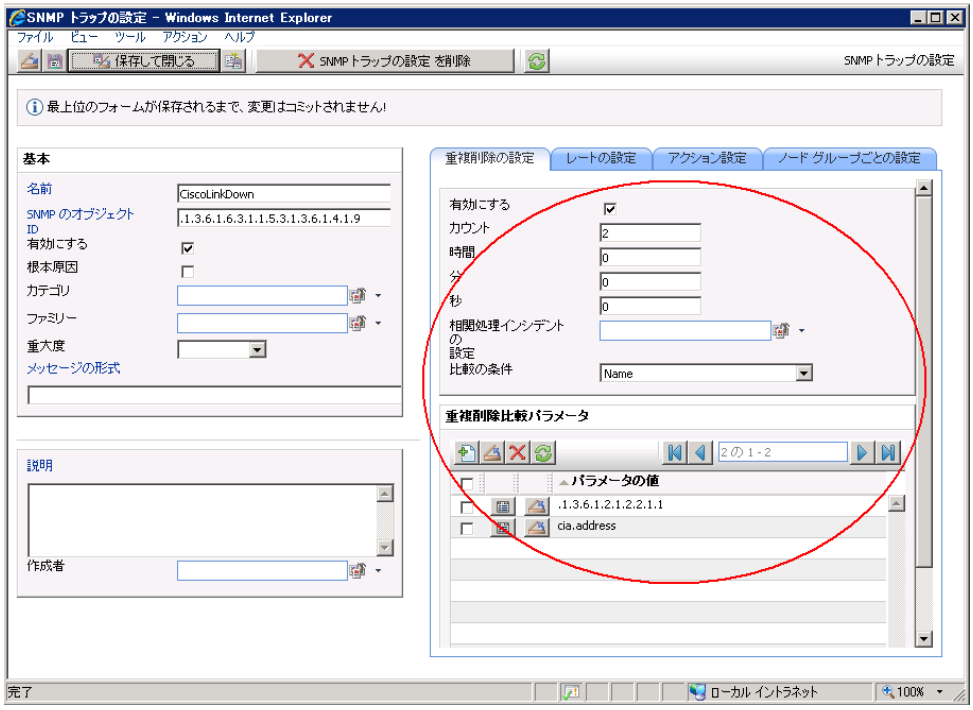
   詳細は、*nnmincidentcfg.ovpl* と *nnmloadmib* のリファレンスページまたは UNIX のマンページを参照してください。

これらの手順では、トラップ型および通知型の MIB エントリをロードするのみです。NNMi は他の MIB 変数を使いません。

4 NNMi コンソールで、**[設定]** ワークスペースから **[インシデントの設定]** を選択します。次に、**[SNMP トラップの設定]** タブを選択します。



5 トラップ表示が NNM での表示と一致するようにカスタマイズします。トラップ設定フォームで、必要に応じてカテゴリを作成できます。



NNMi での強化

6 (オプション)。デフォルト設定 **Severity (重要度)**、**Category (カテゴリ)**、および **Message (メッセージ)** に加えて、デフォルトの **Family (ファミリ)** を設定します。

7 (オプション) 。トラップが **[根本原因インシデント]** ビューに表示されるように、トラップを根本原因として分類します。

NNMi で生成された管理イベント表示のカスタマイズ

NNMi では、イベント設定は簡単になっています。NNMi Causal Engine は NNM よりも簡潔な根本原因を生成します。

NNMi で生成されたインシデントを変更し、NNM アラームと類似した外見にします。たとえば、NNMi `NodeDown` インシデントメッセージを NNM `NodeDown` アラームメッセージに類似するようカスタマイズできます。

NNM から収集

1. NNM で、イベント設定のカスタマイズを決定します。

NNMi に複写

2. NNMi コンソールで、**[設定]** ワークスペースから **[インシデントの設定]** を選択します。次に、**[管理イベントの設定]** タブを選択します。
3. イベント番号ではなく名前で、新しいインシデント設定を見つけます。
4. *オプション*。イベント表示を NNM のイベント表示と一致するようカスタマイズするには、トラップ設定フォームでカテゴリを作成します。
5. デフォルトの **Severity（重要度）**、**Category（カテゴリ）**、および **Message（メッセージ）** 設定に加えて、デフォルトの **Family（ファミリ）** を設定できます。

トラップのブロック / 無視 / 無効化

NNM にはさまざまなレベルのイベント処理が備わっています。

- トラップが `ovtrapd` に入ってくるときにトラップをブロック
- `IGNORE` というラベルのトラップまたはイベントは、処理はするが、保存または表示はしない
- `LOGONLY` というラベルの（相関）イベントを保存および処理するが、表示はしない
- イベントをカテゴリに保存、処理、表示する
- 設定なしに到着するトラップは、`trapd.conf` でフォーマットなしとして Alarm Browser に表示され、データベースに保存されます。

NNMi にはもっと簡単な方法があります。*disabled（無効）* イベントまたはトラップは保存、処理、または表示されません。*enabled（有効）* イベントまたはトラップは完全に保存、処理、表示されます。NNMi に設定がないイベントはブロックされます。

NNM から収集

1. トラップを無視するカスタマイズまたはトラップを `LOGONLY` に設定するカスタマイズを決定します。
2. NNM がトラップフィルタメカニズム（`ovtrapd.conf`、NNM 7.51 では新規）を使用するかどうか調べます。

NNMi に複写

3. NNMi コンソールで、**[設定]** ワークスペースから **[インシデントの設定]** を選択します。受信または表示したくないイベントを見つけ、これらイベントの **[有効化]** チェックボックスをオフにします。
4. 特定の IP アドレスからトラップをブロックするには、以下のファイルを編集し、NNM からのトラップ フィルタリング情報を使用して NNMi をアップデートします。
   - *Windows*: `%NnmDataDir%\shared\nnm\conf\nnmtrapd.conf`
   - *UNIX*: `$NnmDataDir/shared/nnm/conf/nnmtrapd.conf`
5. `nnmtrapconfig.ovpl` コマンドを使用してトラップブロッキングを有効にし、トラップブロッキングのレートとしきい値を設定します。

   このコマンドの使用法の詳細は、*nnmtrapconfig.ovpl* リファレンスページまたは UNIX のマンページを参照してください。

自動処理の設定

NNM から収集

1 NNM 用に設定された自動処理を決定します。

NNMi に複写

2 NNM 管理ステーションの処理スクリプトを NNMi 管理サーバーにコピーします。この場合、ファイルの位置は重要ではありません。

3 NNMi コンソールで、**[設定]** ワークスペースから **[インシデントの設定]** を選択します。

4 自動処理のある NNM イベントごとに、その処理に対応する NNMi インシデントを設定します（**[アクションの設定]** タブ）。

NNM の動作と一致させるために、**[ライフサイクル状態]** を **[登録済み]** に設定します。

NNMi での強化

5 次の NNMi 設定テクニックに注意してください。

- イベント到着時に発生する複数の自動処理を設定できます。
- 他のライフサイクル状態ごとに、1 つまたは複数の追加処理を設定できます（ライフサイクル状態は、**In Progress（進行中）**、**Completed（完了済み）**、**Closed（閉じている）**）。
- NNM より多くのインシデント属性をコマンドに渡せます。
- NNMi がコマンドを実行する前に、別の設定ファイルにコマンドを登録する必要はないので、手順は簡単になっています。

追加（手動）処理の設定

NNM には、Alarms Browser のメニューから利用できるオペレータの操作または追加の操作が用意されています。NNMi コンソールメニューから利用できる URL 操作で NNM 処理をシミュレートすることもできます。

NNM から収集

1 NNM にあるカスタムオペレータ操作を決定します。

NNMi に複写

2 これらのユーザー操作について、URL として利用できるように転送する方法を決定します。

3 NNMi コンソールで、**[設定]** ワークスペースから **[URL アクション]** を選択します。

4 **[新規作成]** をクリックします。

5 アクションについて **[メニューラベル]**、**[一意のキー]**、**[順序]**、**[選択の種類]**、**[完全な URL]** をすべて用意します。

イベント相関処理：イベントの繰り返し

NNM では、イベントを複製するときに、最初のイベントまたは最後のイベントのどちらかを親として使用します。

NNMi では、**[インシデントの参照]** ワークスペースから **[相互関連するインシデントを単純化]** を選択すると、新しい親が作成され、表示されます。最初のイベントが、設定されたビューに表示されます。

NNM から収集

1 `RepeatedEvents` 相関処理が NNM に使われるかどうか調べます。

2 `Repeated` 相互関係が NNM に使われるかどうか調べます。

3 複製が使われているかどうか調べます（`dedup.conf` ファイル）。

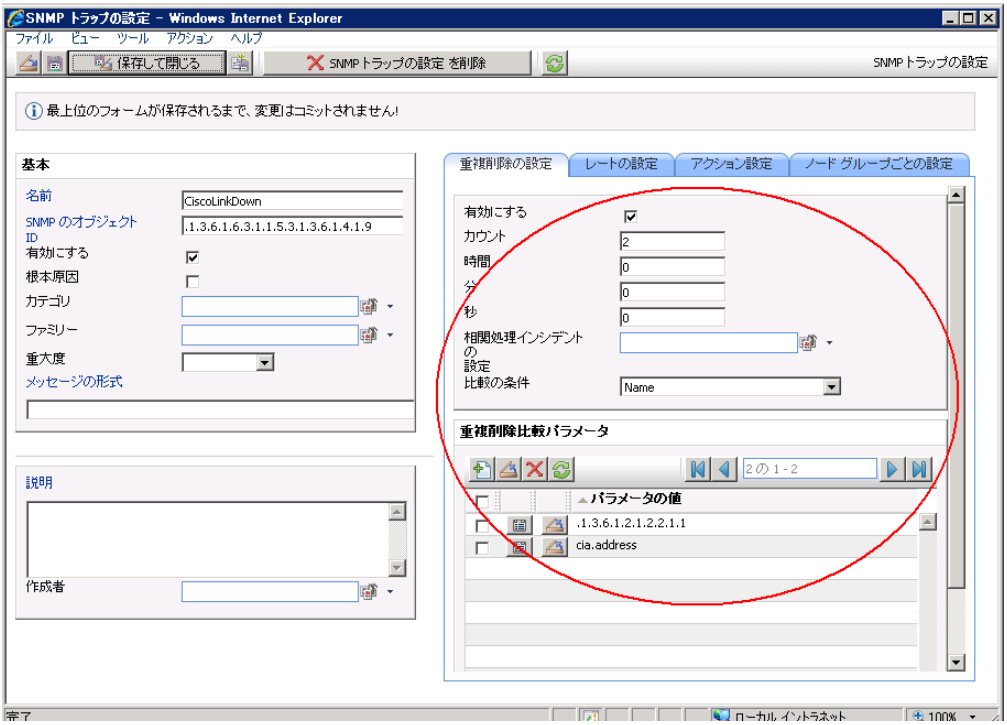
NNMi に複写

4 NNMi コンソールで、**[設定]** ワークスペースから **[インシデントの設定]** を選択します。

5 複製するイベントを開きます。

6 **[複製の設定]** タブを選択し、複製を有効にし、新しい親イベントを選択し、一致基準を定義します。

NNMi での複製には時間の制限がありません。



イベント相関処理：レート計算

NNM では、イベントを複製するときに、最初のイベントまたは最後のイベントのどちらかを親として使用します。

NNMi では、**[インシデントの参照]** ワークスペースから **[相互関連するインシデントを単純化]** を選択すると、新しい親が作成され、表示されます。最初のイベントが設定済みのビューに表示されます。NNMi は、レートの動作を NNM の定期的時間ウィンドウに等しいように保持してきました。

NNM から収集

1 レート相関処理が NNM に使われるかどうか調べます。

NNMi に複写

2 NNMi コンソールで、**[設定]** ワークスペースから **[インシデントの設定]** を選択します。

3 **[管理イベントの設定]** タブを選択します。

4 カウントされるイベント識別子を開きます。

5 **[速度の設定]** タブを選択し、以下を実行します。

   a **[有効化]** を選択してモニタリングを有効にします。

   b カウントウィンドウを設定します。

   c 時刻ウィンドウを設定します（**[時間]**、**[分]**、および **[秒]** の各フィールド）。

   d 新しい親イベントを選択します（**[相関処理インシデントの設定]**）。

e ［**比較基準**］を定義します。

詳細は、NNMi ヘルプの「*管理イベントフォーム*」を参照してください。

イベント相関処理：Pairwise のキャンセル

NNMi では、キャンセルは特定の時刻ウィンドウに制限されません。

NNM から収集

1 NNM で、PairWise のレート相関処理が使われるかどうか調べます。

2 NNM で、過渡的レート相関処理が使われるかどうか調べます。

NNMi に複写

3 NNMi コンソールで、［**設定**］ワークスペースから［**インシデントの設定**］を選択します。

4 ［**Pairwise の設定**］タブを選択します。

5 既存のペアを選択するか、または［**新規作成**］をクリックします。

6 ペアにされたイベント識別子および一致基準を設定します。

詳細は、NNMi ヘルプの「*インシデントを設定する*」を参照してください。

イベント相関処理：スケジュールされたメンテナンス

NNMi では、使用不能ノードのモニタリングを抑制できます。これを行うには、サービス外モードを使います。NNM とは異なり、サービス外メンテナンスを前もってスケジュールすることはできません。手動でオブジェクトを管理対象モードに戻す必要があります。

サービス外モードのデバイスが送信した SNMP トラップは NNMi 内で抑制されます。

組織がスケジュールされたメンテナンス相関処理を使っている場合は、一緒にオフラインになったシステムのリストを使用できます。

NNM から収集

1 `ScheduledMaintenance` 相関処理が NNM に使われるかどうか調べます。

NNMi に複写

2 NNMi コンソールで、［**設定**］ワークスペースから［**ノードグループ設定**］を選択します。

3 **NNM メンテナンスリスト**内のノードのセットごとにノードグループを作成します。ノードグループをビューフィルタとして利用できるように設定します。

4 メンテナンスのときは、NNMi コンソールで［**設定**］ワークスペースから［**ノード**］を選択します。

5 ビューを特定のノードグループにフィルタするには、上端の［**ノードグループフィルタを設定**］セレクタを使用します。

6 全ノードを選択してから、［**アクション**］>［**サービス外**］を選択します。

7 メンテナンスが完了した後、ノードを選択してから、［**アクション**］>［**管理**］を選択します。

NNM 6.x または NNM 7.x と NNMi 8.10 との統合

HP Network Node Manager (NNM) 6.x/7.x の機能を HP Network Node Manager i Software (NNMi) 8.10 と統合することができます。

- NNM 6.x/7.x のイベントを NNMi 8.10 管理サーバーに転送して、NNMi 8.10 インシデントビューを、インシデントライフサイクルを管理するために使用できます。
- いくつかの NNM 6.x/7.x ビューは NNMi 8.10 管理サーバーから開くことができます。

この統合は、新しい NNMi 8.10 製品への移行の割合を管理する場合に便利です。

この統合は、多数の NNM 6.x/7.x 管理ステーションを備えた大規模な管理環境にも便利です。ネットワーク全体を通して NNMi 8.10 の新しい機能を必要としない場合は、NNMi 8.10 を本来のネットワーク管理ツールとして使用しながら若干の NNM 6.x/7.x 管理ステーションを維持することができます。

この章の内容を使用して、サードパーティの製品を NNMi 8.10 と統合することもできます。このサードパーティ製品は、SNMP v1、v2C、または v3 トラップを発生して NNMi 8.10 管理サーバーに送信できることが必要です。

この章では、以下の内容を記載しています。

- イベント転送の設定
- リモートビュー起動の設定
- 統合をテストする
- イベント転送のトラブルシューティング

イベント転送の設定

NNM 6.x/7.x 管理ステーションから NNMi 8.10 管理サーバーへのイベント転送を設定するには、以下の手順を順に行います。

- ステップ 1：NNM 6.x/7.x を、NNMi 8.10 管理サーバーにイベントを転送するように設定する
- ステップ 2：（任意）ノードレベルのフィルタリングを使用してイベント数をさらに削減する
- ステップ 3：NNM 6.x/7.x 管理ステーションを NNMi 8.10 トポロジに追加する
- ステップ 4：（任意）管理ステーション設定を保存する
- ステップ 5：NNM 6.x/7.x インシデント設定を NNMi 8.10 コンソールで確認する

ステップ 1：NNM 6.x/7.x を、NNMi 8.10 管理サーバーにイベントを転送するように設定する

NNM 6.x/7.x 管理ステーションで、NNMi 8.10 管理サーバーに転送したい各々のイベントを設定します。これらのイベントのほとんどは、OpenView Enterprise の下にあります。興味深いイベントとしては、次のようなものがあります。

- OV_Node_Down（OV_Node_Up、Ov_Node_Unknown、その他）
- OV_APA_NODE_DOWN（OV_APA_NODE_Intermittent、その他）
- OV_Station_Critical（OV_Station_Normal、その他）
- OV_Error（OV_Warning、OV_Inform）システムの状態に関する情報
- OV_Message（OV_Popup_Message、その他）

転送が推奨される NNM 6.x/7.x イベントの全一覧については、NNMi コンソールで **［インシデントの設定］** フォームの **［リモート NNM 6.x/7.x イベント設定］** タブのイベント一覧をご覧ください。

推奨およびサポートされる手順：［イベント設定］ウィンドウを使用する

XServer がない場合は、別の手順：trapd.conf を手順で編集するページ 176 を参照してください。

NNM 6.x/7.x イベントを設定して NNMi 8.10 管理サーバーに転送するには、以下の手順に従います。

1 コマンドプロンプトで、次のように入力します。

```
ovw
```

または、xnmtrap をコマンドラインから実行して、手順 3 から続けます。

2 **［オプション］** > **［イベント設定］** をクリックします。

3 **［イベント設定］** ウィンドウで、最上部のウィンドウの **Openview** エンタープライズを選択し、一番下のウィンドウでイベント名をダブルクリックします。

イベントを名前でソートするには、**［ビュー］** > **［ソート］** > **［イベント名］** をクリックします。

4 NNMi 8.10 管理サーバーが転送されたイベントを受信するように指定します。

ベストプラクティス　転送先リストファイルを作成してある場合は、このファイルへの完全パスを［**転送先**］フィールドに入力します。転送先リストファイルの書式については、任意：転送先リストファイルページ 175 を参照してください。

- *Windows*：［**イベントの変更**］ウィンドウの［**転送**］タブで、NNMi 8.10 管理サーバーのホスト名を［**転送先**］フィールドに入力します。

  ［**追加**］をクリックして、［**OK**］をクリックします。

- *UNIX*：［**Event Configurator**］ウィンドウの最下部の［**転送先**］フィールドで、NNMi 8.10 管理サーバーのホスト名を入力します。

➤ ［**転送先**］フィールドが表示されない場合は、ウィンドウの中央の［**イベントの転送**］オプションを選択します。

［**追加**］をクリックして、［**OK**］をクリックします。

5 手順 3 と 手順 4 を、NNMi 8.10 管理サーバーへ転送したいイベントがすべて設定されるまで繰り返します。

6 ［**ファイル**］>［**保存**］をクリックします。

NNM 6.x/7.x はイベント設定の変更を保存して、新しいイベント設定を自動的に再読み込みします。

任意：転送先リストファイル

いくつかのイベントを同じ NNMi 8.10 管理サーバーのグループに転送したい場合は、転送先がリストされたファイルを作成します。

転送先リストファイルに推奨される場所は、以下のとおりです。

- Windows: `%OV_CONF%\nnm8EventForwardDestinations.txt`
- UNIX: `$OV_CONF/nnm8EventForwardDestinations.txt`

転送先リストファイルは、次の書式を持つテキストファイルです。

- 各行は、1 つのノード名かコメント行です。
- コメント行の最初の文字は、# 文字です。

例：

```
# List of destination NNMi 8.10 Management Servers to receive events.
# This list should be small enough that it does not overwhelm the NNMi 8.10 operators.
# In general, the events should be node-related, so that Neighbor Views launched remotely
from the NNMi 8.10 management server are meaningful.
#
system1.domain.com
system2.comain.com
system3.domain.com
```

詳細については、`trapd.conf` マンページを参照してください。

➤ 転送先リストファイルを作成または変更した後で、次のコマンドを実行して再読み込みします。

```
xnmevents -event
```

別の手順：trapd.conf を手順で編集する

XServer がない場合は、以下のファイルで各イベントの FORWARD フィールドを手動で編集できます。

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

NNMi 8.10 管理サーバーを 1 つだけ指定するか、転送先リストファイルを指定します。例：

```
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
```

FORWARD フィールドには、リモートマネージャのリストを含めることもできます。例：

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

trapd.conf ファイルを編集した後で、次のコマンドを実行して NNM にイベント設定を再読み込みさせます。

```
xnmevents -event
```

ステップ 2：（任意）ノードレベルのフィルタリングを使用してイベント数をさらに削減する

NNM 6.x/7.x では、特定のイベントにノードリストを設定できます。ノードリストがあるとき、NNM 6.x/7.x 管理ステーションに入ってくるイベントがイベント設定に一致するのは、イベントソースがノードリスト内にある場合のみです。そのため、イベントが NNMi 8.10 管理サーバーに転送されるのは、イベントソースがノードリスト内にある場合のみになります。ノードリストの典型的な使用事例は、重要ノードから特定のイベントのみを NNMi 8.10 管理サーバーへ転送することです。

NNM 6.x/7.x でノードリストを作成する場合の詳細は、ovtrapd.conf マンページで sources_list に関する情報を参照してください。

ステップ 3：NNM 6.x/7.x 管理ステーションを NNMi 8.10 トポロジに追加する

NNM 6.x/7.x 管理ステーションを NNMi 8.10 トポロジに加えて、NNM 6.x/7.x 管理ステーションが停止したときに NNMi 8.10 管理サーバーがインシデントを受信するようにします。

NNM 6.x/7.x 管理ステーションがまだ NNMi 8.10「**ノード**」インベントリビューにない場合、この管理ステーションを検出シードに追加して、検出されるまでお待ちください。

ノードを検出シードに追加する方法の詳細は、NNMi ヘルプの*「ネットワークの検出」*を参照してください。

ステップ 4：（任意）管理ステーション設定を保存する

新しい設定を保存するには、次のコマンドを実行します。

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

次のコマンドを実行すると、バックアップを後でインポートできます。

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

これらのコマンドの詳細は、それぞれのリファレンスページか UNIX のマンページを参照してください。

ステップ 5 : NNM 6.x/7.x インシデント設定を NNMi 8.10 コンソールで確認する

NNM 6.x/7.x から転送するイベントが（インシデントとして）NNMi 8.10 で設定されていることを確認します。

NNMi 8.10 には、そのまま使用できるインシデント設定が多数あります。これらのインシデント設定を表示するには、以下の手順に従います。

1. ワークスペース用ナビゲーションパネルから、**［設定］**ワークスペースを選択します。
2. **［インシデントの設定］**フォームを開きます。
3. **［リモート NNM 6.x/7.x イベント設定］**タブをクリックします。

   このタブには、そのまま使用できるインシデント設定が表示されます。インシデントを開くと、タイプが**「リモートで生成」**であることが表示されるはずです。

NNM 6.x/7.x 管理ステーションからの転送を設定したイベントの 1 つまたは複数が**［リモート NNM 6.x/7.x イベント設定］**タブにリストされていない場合は、紛失しているイベントごとに新しいインシデント設定を追加します。新規インシデントの設定方法の詳細は、NNMi ヘルプの*「インシデントを設定する」*を参照してください。

NNM 6.x/7.x のインシデントカテゴリは、NNMi 8.10 のそれとは異なります。NNM 6.x/7.x アラームカテゴリと NNMi 8.10 インシデントカテゴリとの間の関係の詳細は、カテゴリのマッピングを参照してください。

カテゴリのマッピング

NNM 6.x/7.x では、設定済みのアラームカテゴリは以下のとおりです。

- エラーアラーム
- しきい値アラーム
- ステータスアラーム
- 設定アラーム
- アプリケーションアラートアラーム

NNMi 8.10 では、設定済みのインシデントカテゴリは以下のとおりです。

- アカウント
- アプリケーションステータス
- 設定
- 障害
- パフォーマンス
- セキュリティ
- ステータス

HP では、NNM 6.x/7.x アラームカテゴリから NNMi 8.10 インシデントカテゴリへの以下のマッピングをお勧めしています。

NNM 6.x/7.x アラームカテゴリ	NNMi 8.10 インシデントカテゴリ
エラーアラーム	アプリケーションステータス
しきい値アラーム	パフォーマンス
ステータスアラーム	ステータス
設定アラーム	設定
アプリケーションアラートアラーム	アプリケーションステータス

リモートビュー起動の設定

NNMi 8.10 管理サーバーが NNM 6.x/7.x ビューを NNMi 8.10 管理サーバーで表示するように設定するには、以下の手順を順に行います。

- ステップ 1：Java Run-Time Environment (JRE) のインストール
- ステップ 2：NNMi 8.10 で NNM 6.x/7.x 管理ステーションエンティティを作成する
- ステップ 3：（任意）追加の NNM 6.x/7.x ビューを設定する

ステップ 1：Java Run-Time Environment (JRE) のインストール

NNMi 8.10 には Java Run-Time Environment (JRE) の要件はありませんが、NNM 6.x/7.x ビューでは JRE のバージョン 1.4.2.xx が必要になります。必要な JRE をダウンロードしてインストールするには、次のサイトにアクセスします。

http://java.sun.com/j2se/1.4.2/download.html

NNM 7.53 は、新しい JRE バージョンをサポートしていません。詳細については、NNM 7.53 リリースノートを参照してください。

ステップ 2：NNMi 8.10 で NNM 6.x/7.x 管理ステーションエンティティを作成する

NNMi 8.10 管理サーバーを設定して、NNM 6.x/7.x 管理ステーションから受信したイベントを NNMi 8.10 のエンティティに関連付けます。この設定により、NNMi 8.10 管理サーバーからの NNM 6.x/7.x 動的ビューの起動が可能になります。たとえば、NNMi 8.10 で表示されている My7xSystem からノード停止を選択して、My7xSystem に戻る URL を起動できます。

大切なのは、NNM 6.x/7.x 管理ステーションにより送信されたイベントでエンコーディングされているアドレスに一致するプライマリアドレスを使用することです。このアドレスがよくわからない場合は、カスタムインシデント属性の **RemoteSenderAddress** で NNM 6.x/7.x 管理ステーションから転送されたインシデントについて調べてください。

NNM 6.x/7.x 管理ステーションを設定するには、以下の手順に従います。

1 NNMi 8.10 コンソールのワークスペース用ナビゲーションパネルで、**[設定]** ワークスペースを選択します。

2 **[管理ステーション]** フォームを開きます。

3 **[新規作成]** をクリックします。

4 以下の情報を入力します。

- **名前** — この設定によって表示される NNM 6.x/7.x 管理ステーションの識別名。
- **NNM バージョン** — 設定している管理ステーションの NNM バージョン（6.x または 7.x）。
- **アドレス** —NNM 6.x/7.x 管理ステーションの IP アドレス。この IP アドレスは、NNMi 8.10 管理サーバーから到達可能である必要があります。IP アドレスは、以下のどちらかの方法で見つけることができます。
  - — ovaddr を NNM 6.x/7.x 管理ステーションのコマンドラインで実行します。
  - — NNM 6.x/7.x 管理ステーションから転送されたインシデントのカスタムインシデント属性 (CIA) を判別します。

この方法が有効なのは、イベント転送の設定ページ 174 で説明した手順をすでに済ませており、設定したイベントが NNM 6.x/7.x 管理ステーションで発生して NNMi 8.10 管理サーバーへ転送されている場合のみです。

- **ovas（OV アプリケーションサーバー）ポート** — 設定している NNM 7.x 管理ステーションの OpenView アプリケーションサーバー (ovas) のポート番号。NNM 7.x 管理ステーションでは、このポート番号は通常は 7510 です。

ovas ポートは、Extended Topology アドオンを備えた NNM 6.x にも適用されます。

- **Web サーバーポート** — 設定している NNM 6.x/7.x 管理ステーションの Web サーバーのポート番号。
  - — Windows OS の NNM 6.x 管理ステーションでは、このポート番号は通常は 80 です。
  - — UNIX OS の NNM 6.x 管理ステーションでは、このポート番号は通常は 3443 です。
  - — あらゆる OS での NNM 7.x 管理ステーションでは、このポート番号は通常は 3443 です。
- **説明** — 設定している NNM 6.x/7.x 管理ステーションの説明。

5 **[保存して閉じる]** をクリックします。

ステップ 3：（任意）追加の NNM 6.x/7.x ビューを設定する

以下の URL は、既定では追加されていません。これらの URL はいずれも NNM 6.x/7.x 配備に追加できます。

選択を必要としない URL

- MIB ブラウザの URL 例：

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

- Report Presenter の URL 例：

http://192.168.1.*xxx*:3443/OvCgi/nnmRptPresenter.exe

- Topology Summary の URL 例：

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

- SNMP Data Presenter（MIB フォーム / テーブル contrib. graphs）：

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

- OV Launcher の URL 例：

http://system.yourcompany.com:3443/OvCgi/ovlaunch.exe

- jovw の URL 例：

  （Web ベースの ovw は、ovw セッションを実行中であることが必要です。実行していないと、エラーダイアログ「ovw がホストに見つかりません ...」が表示されます）：

http://system.yourcompany.com:3443/OvCgi/jovw.exe

➤ この URL は、次のようなオプションでのコンテキストノードとマップ名を使用できます。jovw.exe?mapName=default&ObjectName=10.1.12.33

- ovalarm の URL 例：

http://system.yourcompany.com:3443/OvCgi/ovalarm.exe

- トポロジ詳細を要求するフォーム（名前、IP アドレス、物理アドレス UUID、OvwId でノードを入力します）：

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

選択を必要とする URL

- ovwId を使用しているノード詳細：

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

- UUID を使用しているノード詳細：

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

統合をテストする

NNM 6.x/7.x と NNMi 8.10 管理サーバーとの統合を正しく設定したことを確認するには、以下の手順の 1 つ以上を必要に応じて行います。

- テスト 1：イベント転送を確認する
- テスト 2：NNM 6.x/7.x 動的ビューを NNMi 8.10 から起動する

テスト 1：イベント転送を確認する

NNM 6.x/7.x 管理ステーションから NNMi 8.10 管理サーバーへのイベント転送を確認するには、以下の手順に従います。

1 NNM 6.x/7.x 管理ステーションで、転送されるイベントの 1 つを発生する状況をつくります。

最も簡単な方法は、sendMsg.ovpl コマンドを NNM 6.x/7.x 管理ステーションで実行することです。このコマンドの実行方法については、sendMsg.ovpl を参照してください。

もう 1 つの方法は、NNM 6.x/7.x システムでネットワーク障害を発生させるかシミュレートすることです。ノード / インタフェースの停止 / 稼働イベントを発生させるページ 181 を参照してください。

2 NNMi 8.10 コンソールのワークスペース用ナビゲーションパネルで、**[インシデント]** ワークスペースを選択します。

3 **[NNM 6.x/7.x イベント]** を選択します。

NNM 6.x/7.x 管理ステーションから発生させたイベントがこのビューで見えるはずです。

また、**nnmdumpevents -t** を NNMi 8.10 管理サーバーで実行して、NNMi 8.10 管理ステーションが受信したイベントのリストを見ることもできます。

ノード / インタフェースの停止 / 稼働イベントを発生させる

ノード停止イベントの原因となりうる偽のインタフェース停止イベントを発生させるには、次のコマンドを実行します。

ovtopofix -S Down <*IPADDR*>

<IPADDR> は、NNM 6.x/7.x 管理ステーションで検出されたインタフェースのうちの 1 つの IP アドレスです。

ovtopofix コマンドを、ノードの IP インタフェースごとに 1 回実行します。使用する IP アドレスを決定するには、次のコマンドを実行します。

ovtopodump > topology.txt

topology ファイルで NODES の語を見つけ出し、それから NNM 6.x/7.x 管理ステーションのエントリを探します。例：

```
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
```

すべての IP インタフェースのステータスが Critical（危険域）になると、NNM はノードを停止として表示します。

また、NNM 6.x/7.x 管理ステーションのノード名またはトポロジ ID を ovtopofix コマンドの最後の引数として指定できます。他のオプションについては、ovtopofix マンページを参照してください。

- インタフェースおよびノード停止イベントを発生させた後で、次のコマンドを IP インタフェースごとに 1 回実行して、インタフェース稼働イベントとノード稼働イベントを発生させます。

  ovtopofix -S Up <*IPADDR*>

- テストしているイベント（この場合は OV_IF_Up/OV_IF_Down で、それぞれ .1.3.6.1.4.1.11.2.17.1.0.58916866 と .1.3.6.1.4.1.11.2.17.1.0.58916867）が NNMi 8.10 管理サーバーに転送されるよう設定されていることを確認します。また、これらのイベントのルールの書式を整形したことも確認します。整形されていないと、インシデントビューで不明として表示されます。

sendMsg.ovpl

sendMsg.ovpl コマンドを実行すると、OV_Message イベントを発生させることができます。

例：

- Windows:

```
%OV_CONTRIB%\NNM\sendMsg\sendMsg.ovpl "" "Test from `hostname` on `date`"
```

- UNIX:

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

sendMsg.ovpl コマンドを実行するたびに、NNM では sendMsg.ovpl コマンドラインで追加したテキストを含む OV_Message イベントが発生します。例：

```
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
```

このイベントは、NNM 6.x/7.x 管理ステーションの**「すべてのアラーム」**ブラウザで表示されます。

ベストプラクティス　新しいアラームを見分けやすくするには、sendMsg.ovpl コマンドを実行する前に**「すべてのアラーム」**ブラウザですべてのアラームを削除します。

- デフォルトでは、OV_Message インシデントは NNMi 8.10 では設定されていません。このテストを実行するためには、NNM 6.x/7.x で OV_Message イベントの転送の設定が済んでおり、NNMi 8.10［**インシデントの設定**］フォームで OV_Message インシデントの設定が済んでいる必要があります。

NNM 6.x/7.x システムへのトラップをテストする

NNM 6.x/7.x がトラップを転送するよう設定したら、転送されているトラップが受信されているのも見えるはずです。

次のようなコマンドを使用して、NNM 6.x/7.x 管理ステーションで手動でトラップを発生させることもできます。

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

- この例では、SNMP_Link_Down トラップが発生します。転送されるように設定したトラップのイベントオブジェクト識別名を使用してください。

詳細については、*snmptrap* マンページを参照してください。ホスト名は NNM 6.x/7.x システムの名前です。

テスト 2 : NNM 6.x/7.x 動的ビューを NNMi 8.10 から起動する

1 NNMi 8.10 コンソールで、設定した NNM 6.x/7.x 管理ステーションを開きます。

［**アクション**］メニューに以下のアクションがあります。

- NNM 6.x/7.x ホームベース
- NNM 6.x/7.x ovw
- NNM 6.x/7.x MIB ブラウザ
- NNM 6.x/7.x Launcher
- NNM 6.x/7.x アラーム

2 それぞれのビューを［**アクション**］メニューから開きます。

イベント転送のトラブルシューティング

表示されるはずの NNM 6.x/7.x イベントが **NNM 6.x/7.x イベント**ビューで表示されない場合は、以下の手順に従って問題を解決してください。

1 NNM 6.x/7.x 管理ステーションで、次のコマンドを実行します。

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

ここで ***<n>*** は、イベント履歴をさかのぼる分時間を指定します。たとえば、*n* の値が 1 の場合、ovdumpevents コマンドは NNM 6.x/7.x 管理ステーションで最後の *1* 分間に発生したイベントを表示します。

2 予想したイベントが ovdumpevents 出力に含まれていない場合、そのイベントは発生しませんでした。この状況のトラブルシューティングに関する情報は、NNM 6.x/7.x のマニュアルを参照してください。

3 手順 1 を、予想したすべてのイベントが NNM 6.x/7.x 管理ステーションの ovdumpevents 出力に含まれるまで繰り返してください。

4 NNMi 8.10 管理サーバーで、次のコマンドを実行します。

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

ここで ***<n>*** は、イベント履歴をさかのぼる分時間を指定します。たとえば、*n* の値が 1 の場合、nnmdumpevents コマンドは NNMi 8.10 管理サーバーで最後の *1* 分間に発生したイベントを表示します。

5 nnmdumpevents 出力に含まれるはずなのに含まれていない各イベントについて、そのイベントの設定を NNM 6.x/7.x 管理ステーションの［**Event Configurator**］ウィンドウで確認します。

- ［**イベントの転送**］オプションが選択されていることを確認します。
- NNMi 8.10 管理サーバーの名前かまたは IP アドレスを、**「転送されるイベントの転送先」**リストで確認します。

詳細については、ステップ 1 : NNM 6.x/7.x を、NNMi 8.10 管理サーバーにイベントを転送するように設定するページ 174 を参照してください。

6 手順 5 を、予想したすべてのイベントが NNMi 8.10 管理サーバーの nnmdumpevents 出力に含まれるまで繰り返してください。

7 NNMi 8.10 コンソールで、**「NNM 6.x/7.x イベント」**インシデントビューを調べます。結果が予想と異なる場合は、**［インシデントの設定］**フォームの**［リモート NNM 6.x/7.x イベント設定］**タブでインシデント設定を確認してください。

統合とプラグイン

この項では以下の章について説明します。

- HP Operations Manager
- HP Universal Configuration Management Database（HP ユニバーサル設定管理データベース）
- AlarmPoint
- nGenius Performance Manager
- NNM iSPI for Performance

HP Operations Manager

HPOM (HP Operations Manager) には、包括的なイベント管理、先を見越したパフォーマンスモニタリング、および管理オペレーティングシステム、ミドルウェア、およびアプリケーションインフラストラクチャ用の自動アラート、レポート、およびグラフが備わっています。HPOM は広範囲のソースから 1 つのビューにイベントを統合します。

HPOM の購入については、HP 販売員にお尋ねください。

この章では、以下の内容を記載しています。

- HP NNMi–HPOM 統合
- HP NNMi–HPOM 統合の有効化
- HP NNMi–HPOM 統合の使用法
- NNMi–HPOM 統合設定の変更
- HP NNMi–HPOM 統合の無効化
- HP NNMi–HPOM 統合のトラブルシューティング
- HP NNMi–HPOM Integration Configuration フォームリファレンス
- HP NNMi–HPOM 統合ログファイルリファレンス

HP NNMi–HPOM 統合

NNMi HPOM 統合は NNMi インシデントを HPOM アクティブメッセージブラウザに転送します。統合によって、NNMi と HPOM の間でインシデントが同期されます。HPOM 内部から NNMi コンソールにアクセスすることもできます。

NNMi HPOM 統合は、「多対多」の配置をサポートします。各 NNMi 管理サーバーは、複数の HPOM 管理サービスにインシデントを転送できます。同じように、各 HPOM 管理サーバーは複数の NNMi 管理サーバーからインシデントを受信できます。統合では、インシデントの一意の識別子を解釈して、ソース NNMi 管理サーバーを決定します。

NNMi–HPOM 統合は以下のコンポーネントで構成されます。

- **NNMi–HPOM 統合モジュール**

  NNMi–HPOM 統合モジュールは、NNMi から HPOM にインシデントを転送します。これは NNMi 管理サーバーにインストールおよび設定されます。

- **HP Operations Manager インシデント Web サービス**

  HPOM は、HP Operations Manager インシデント Web サービス (IWS) を使用して、NNMi から転送されたインシデントを受信します。

- **コンテキストアクセス用の HPOM アプリケーション NNMi コンソール**

  HPOM には、NNMi コンソールのフォーム、ビュー、およびツールにアクセスするアプリケーションがあります。たとえば、HPOM アクティブメッセージブラウザから直接 NNMi インシデントを開けます。特定のアプリケーションによって、NNMi コンソールが開かれるコンテキストが決定されます。アプリケーション使用する前に、そのアプリケーションを設定する必要があります。

値

NNMi–HPOM 統合には、ネットワーク管理、システム管理、およびアプリケーション管理ドメイン用の HPOM アクティブメッセージブラウザ内にイベント統合体が用意されているので、HPOM ユーザーは潜在的なネットワーク問題を検出および調査できます。

統合の主要な機能は次のとおりです。

- NNMi から HPOM への自動インシデント転送。
  - 転送されたインシデントは HPOM アクティブメッセージブラウザに表示されます。
  - NNMi がどのインシデントを転送するか制限するフィルタを作成できます。
- 次表に説明する NNMi と HPOM の間のインシデント更新の同期。

トリガ	結果
HPOM では、メッセージは肯定応答されます。	NNMi では、対応するインシデントのライフサイクル状態は［閉じている］に設定されます。
HPOM では、メッセージは肯定応答されません。	NNMi では、対応するインシデントのライフサイクル状態は［登録済み］に設定されます。
NNMi では、インシデントのライフサイクル状態は［閉じている］に設定されます。	HPOM では、対応するメッセージは肯定応答されます。
NNMi では、インシデントのライフサイクル状態は［閉じている］から他の状態に変更されます。	HPOM では、対応するメッセージは肯定応答されません。

- HPOM から NNMi コンソールへのアクセス。
  - HPOM ユーザーは、選択したメッセージのコンテキストで NNMi インシデントフォームを開けます。
  - HPOM ユーザーは、選択したメッセージとノードのコンテキストで、NNMi ビュー（たとえば、Layer 2 Neighbor ビュー）を起動できます。

— HPOM ユーザーは、選択したメッセージとノードのコンテキストで、NNMi ツール（たとえば、ステータスポーリング）を起動できます。

— HPOM が複数の NNMi 管理サーバーの NNMi インシデントを統合するとき、正しい NNMi 管理サーバーにアクセスするために、各インシデントの一意の識別子が解釈されます。

サポートされるバージョン

この章の情報は、以下の製品バージョンに当てはまります。

- Windows 用 HPOM バージョン 8.10 以降
- UNIX 用 HPOM バージョン 8.30 以降
- NNMi バージョン 8.10 以降

HP NNMi HPOM for NNMi 8.0x（NNMi 8.03 以降）の設定方法の詳細は、ご使用の NNMi のバージョンに対応する『*NNMi Deployment Guide（NNMi 導入ガイド）*』の「*Integration with HP Operations Manager（HP 操作マネージャとの統合）*」の章を参照してください。

NNMi と HPOM は同じコンピュータにインストールできません。これら 2 つの製品は、以下の構成のどちらかである異なるコンピュータにインストールする必要があります。

- 異なるオペレーティングシステム。たとえば、NNMi 管理サーバーは Linux システムであり、HPOM 管理サーバーは Windows システムです。
- 同じオペレーティングシステム。たとえば、NNMi 管理サーバーが Windows システムであり、HPOM 管理サーバーが 2 番目の Windows システムです。

サポートされているハードウェアプラットフォームおよびオペレーティングシステムの最新情報については、両方の製品の対応マトリックスを参照してください。

ドキュメント

この章では、HPOM と通信するように NNMi を設定する方法について説明します。

HPOM ドキュメントには、NNMi と通信するように HPOM を設定する方法の説明が記載されています。HP NNMi–HPOM 統合の使用方法の説明もあります。

- Windows 用 HPOM の詳細は、HPOM オンラインヘルプの HP NNMi アダプタ情報を参照してください。
- UNIX 用 HPOM の詳細は、『*HP NNMi HPOM Integration for HP Operations Manager User's Guide（HP 操作マネージャ用 HP NNMi HPOM 統合ユーザーガイド）*』を参照してください。

HP NNMi–HPOM 統合の有効化

このセクションでは、NNMi HPOM 統合を有効にする手順について説明します。統合に組み入れる各 NNMi 管理サーバーと各 HPOM 管理サーバーについては、ご使用の HPOM バージョン用手続きの該当するステップを完了してください。

Windows 用 HPOM

1 NNMi 管理サーバーで、HPOM への NNMi インシデント転送を設定します。

   a NNMi コンソールで、**[HP NNMi HPOM 統合設定]** フォーム（**[統合モジュール設定] > [HPOM]**）を開きます。

   b **[統合を有効化]** チェックボックスをオンにし、フォームの残りのフィールドをアクティブにします。

   c NNMi 管理サーバーへの接続情報を入力します。

   これらのフィールドの詳細は、NNMi 管理サーバー接続ページの 198 を参照してください。

   d HPOM 管理サーバーへの接続情報を入力します。

   これらのフィールドの詳細は、HPOM Management Server Connection（HPOM 管理サーバー接続）ページの 199 を参照してください。

   e NNMi が複数の HPOM 管理サーバーにインシデントを転送するようにするには、**[別の OM サーバーを追加]** をクリックしてから、次の HPOM 管理サーバー情報を HPOM のフィールドに入力します。

   最初のサーバーの情報が **[追加の HPOM サーバー]** リストに表示されます。

   f フォームの下端の **[送信]** をクリックします。

   新しいウィンドウにステータスメッセージが表示されます。HPOM サーバー接続に問題があることがメッセージに示されている場合は、**[HP NNMi HPOM 統合設定]** フォームを再び開き（またはメッセージウィンドウの **[ALT] + [LEFT ARROW]** を押し）、エラーメッセージのテキストが指示するように HPOM 管理サーバー接続用の値を調整します。

2 NNMi 管理サーバーで、次のように、統合をカスタマイズします。

   a NNMi コンソールで、**[HP NNMi HPOM 統合設定]** フォーム（**[統合モジュール設定] > [HPOM]**）を開きます。

   b 以下のフィールドに値を入力します。

      — **転送のみ**

      — **保持期間（分単位）**

      — **インシデントフィルタ**

      これらのフィールドの詳細は、統合動作ページの 200 を参照してください。

   c フォームの下端の **[送信]** をクリックします。

3 HPOM で、Windows 用 HPOM オンラインヘルプの「*Configure the NNMi Server Name and Port (NNMi サーバー名とポートの設定)*」に説明してあるように、NNMi 管理サーバーに接続する NNMi アダプタを設定します。

4 HPOM で、各 NNMi ノードの管理対象ノードを作成します。このノードは、この HPOM 管理サーバーに転送される NNMi インシデント内でソースノードとして指名されます。また、HPOM 管理サーバーにインシデントを転送する各 NNMi 管理サーバーの管理対象ノードも作成します。

   この代わりに、転送される全 NNMi インシデントを取得する 1 つの外部ノードを作成することもできます。

詳細は、HPOM オンラインヘルプの「*Configuring NNMi Server Nodes (NNMi サーバーノードの設定)*」を参照してください。

NNMi インシデントソースノード用の HPOM 管理対象ノードを設定しないと、HPOM 管理サーバーは、そのノードに関する全インシデントを破棄します。

5 HPOM で、次のように、NNMi インシデント用のカスタムメッセージ属性をアクティブなメッセージブラウザに追加します。
   a ブラウザで、いずれかの列見出しを右クリックし、**[オプション]**をクリックします。
   b **[カスタムメッセージ属性入力]** リストで、属性を選択し、**[追加]** をクリックします。
      — NNMi インシデントのカスタムメッセージ属性の先頭は nnm というテキストです。
      — NNMi インシデントの最も重要な属性は次のとおりです。

```
nnm.assignedTo
nnm.category
nnm.emittingNode.name
nnm.source.name
```

      — カスタムメッセージ属性がメッセージブラウザに表示される順序を変更するには、列見出しを新しい場所にドラッグします。
6 HPOM で、NNMi ソースノードを HP NNMi Web ツールグループと関連付けして、NNMi ビューのコンテキスト起動を有効にします。

   詳細は、HPOM オンラインヘルプの「*Enabling HP NNMi Web Tools in the By Node Group (By Node グループで HP NNMi Web ツールを有効化)*」を参照してください。

UNIX 用 HPOM

1 次のように、UNIX 管理サーバー用に HPOM を準備します。
   a 「*HP Operations Manager Incident Web Service Integration Guide (HP 操作マネージャインシデント Web サービス統合ガイド)*」に説明してあるように、UNIX 管理サーバー用の HPOM に HP Operations Manager インシデント Web サービス (IWS) をインストールします。
   b UNIX 管理サーバー用の HPOM で、最新の HPOM 統合パッチをインストールします。パッチは次の URL で入手できます。

      http://h20230.www2.hp.com/selfsolve/patches

2 NNMi 管理サーバーで、HPOM への NNMi インシデント転送を設定します。
   a NNMi コンソールで、**[HP NNMi–HPOM 統合設定]** フォーム（**[統合モジュール設定]** > **[HPOM]**）を開きます。
   b **[統合を有効化]** チェックボックスをオンにし、フォームの残りのフィールドをアクティブにします。
   c NNMi 管理サーバーへの接続情報を入力します。

      これらのフィールドの詳細は、NNMi 管理サーバー接続ページの 198 を参照してください。
   d HPOM 管理サーバーへの接続情報を入力します。

      これらのフィールドの詳細は、HPOM Management Server Connection（HPOM 管理サーバー接続）ページの 199 を参照してください。

e NNMi が複数の HPOM 管理サーバーにインシデントを転送するようにするには、**[別の OM サーバーを追加]** をクリックしてから、次の HPOM 管理サーバー情報を HPOM のフィールドに入力します。

最初のサーバーの情報が **[追加の HPOM サーバー]** リストに表示されます。

f フォームの下端の **[送信]** をクリックします。

新しいウィンドウにステータスメッセージが表示されます。HPOM サーバー接続に問題があることがメッセージに示されている場合は、**[HP NNMi–HPOM 統合設定]** フォームを再び開き（またはメッセージウィンドウの **[ALT]** + **[LEFT ARROW]** を押し）、エラーメッセージのテキストが指示するように HPOM 管理サーバー接続用の値を調整します。

3 NNMi 管理サーバーで、次のように、統合をカスタマイズします。

a NNMi コンソールで、**[HP NNMi–HPOM 統合設定]** フォーム（**[統合モジュール設定]** > **[HPOM]**）を開きます。

b 以下のフィールドに値を入力します。

— **Forward Only（転送のみ）**

— **Holding period (minutes)（保持期間（分単位））**

— **Incident Filter（インシデントフィルタ）**

これらのフィールドの詳細は、統合動作ページの 200 を参照してください。

c フォームの下端の **[送信]** をクリックします。

4 HPOM で、各 NNMi ノードの管理対象ノードを作成します。このノードは、HPOM 管理サーバーに転送される NNMi インシデント内でソースノードとして指名されます。また、HPOM 管理サーバーにインシデントを転送する各 NNMi 管理サーバーの管理対象ノードも作成します。

この代わりに、転送される全 NNMi インシデントを取得する 1 つの外部ノードを作成することもできます。

詳細は、「*HP Operations Manager for UNIX Administrator's Reference（UNIX 管理者リファレンスの HP 操作マネージャ）*」を参照してください。

NNMi ソースノード用の HPOM 管理対象ノードを設定しないと、HPOM 管理サーバーは、そのノードに関する全インシデントを破棄します。

5 HPOM で、次のように、NNMi インシデント用のカスタムメッセージ属性をアクティブなメッセージブラウザに追加します。

a ブラウザで、いずれかの列見出しを右クリックし、**[メッセージブラウザ列のカスタマイズ]** をクリックします。

b **[カスタム]** タブで、**[使用可能なカスタムメッセージ属性]** から選択し、**[OK]** をクリックします。

— NNMi インシデントのカスタムメッセージ属性の先頭は nnm というテキストです。

— NNMi インシデントの最も重要な属性は次のとおりです。

```
nnm.assignedTo
nnm.category
nnm.emittingNode.name
nnm.source.name
```

— カスタムメッセージ属性がメッセージブラウザに表示される順序を変更するには、列見出しを新しい場所にドラッグします。

6 HPOM 管理サーバーで、NNMi コンソールにアクセスするために HPOM アプリケーションの準備をします。

a *必須*。NNMi アプリケーションの基本セットをインストールします。

b *オプション*。追加の NNMi アプリケーションをインストールします。

詳細は、『*HP NNMi HPOM Integration for HP Operations Manager User's Guide (HP 操作マネージャ用 HP NNMi HPOM 統合ユーザーガイド)*』の NNMiHPOM 統合のインストールおよび構成についての章を参照してください。

HP NNMi–HPOM 統合の使用法

図 11 に、NNMi コンソールで応答しないインシデントのアドレスを示します。**[ソースオブジェクト]** 列と **[メッセージ]** 列の情報が状況を記述しています。

図 11 NNMi コンソールで応答しないインシデントのアドレス

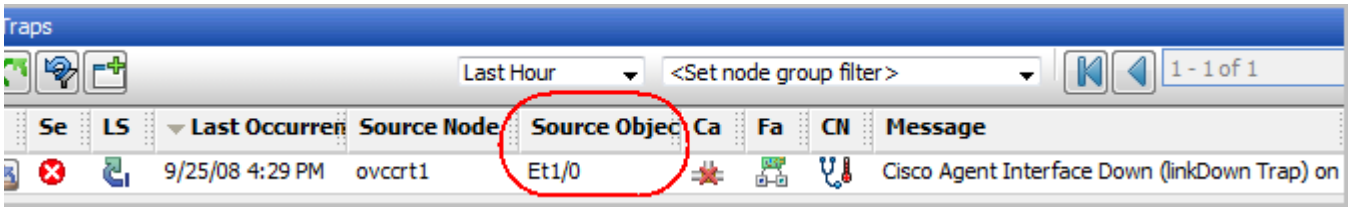


図 12 は、Windows 用 HPOM で受信された NNMi インシデントを示します。図 13 は、UNIX 用 HPOM で受信された NNMi インシデントを示します。**[nnm.source.name]** 列および **[テキスト]** 列は、NNMi コンソールの **[ソースオブジェクト]** 列および **[メッセージ]** 列と同等です。

191 ページの手順 5（Windows 用 HPOM）および 192 ページの手順 5（UNIX 用 HPOM）の説明のように、**[nnm.source.name]** カスタムメッセージ属性列の表示を有効にする必要があります。

図 12 Forwarded Incident in HPOM for Windows（Windows 用 HPOM で転送済みのインシデント）

図 13 Forwarded Incident in HPOM for UNIX（UNIX 用 HPOM で転送済みのインシデント）

NNMi HPOM 統合の使用の詳細は、HPOM ドキュメントを参照してください。

- Windows 用 HPOM の詳細は、HPOM オンラインヘルプの「*NNMi Web Tools (NNMi Web ツール)*」を参照してください。
- UNIX 用 HPOM の詳細は、『*HP NNMi–HPOM Integration for HP Operations Manager User's Guidee (HP 操作マネージャ用 HP NNMi HPOM 統合ユーザーガイド)*』の NNMi アプリケーションの使用法の章を参照してください。

NNMi ヘルプには、この統合によって NNMi コンソールに追加された新規機能の説明は記載されていません。NNMi の各オペレータに、オペレータの責任に関する統合情報を与えてください。

HPOM メッセージブラウザで、転送済み NNMi インシデントの詳細はカスタムメッセージ属性として入手できます。

NNMi–HPOM 統合設定の変更

1. NNMi コンソールで、**[HP NNMi–HPOM 統合設定]** フォーム（**[統合モジュール設定] > [HPOM]**）を開きます。
2. 該当するように値を変更します。
   - Incident Filter リストと Additional HPOM Servers リストのエントリの構文を知っていると、直接にエントリを変更できます。
   - リスト項目の構文を知らない場合は、そのエントリを削除し、再入力します。

   このフォームのフィールドの詳細は、HP NNMi–HPOM Integration Configuration フォームリファレンスページの 198 を参照してください。
3. フォームの上端の**[統合を有効化]**チェックボックスがオンであることを確認し、フォームの下端の **[送信]** をクリックします。

変更はただちに有効になります。`ovjboss` を再起動する必要はありません。

HP NNMi–HPOM 統合の無効化

すべての HPOM 管理サーバーについて

NNMi インシデントのすべての HPOM 管理サーバーへの転送を継続停止にするには、次のステップを実行します。

1. NNMi コンソールで、**[HP NNMi–HPOM 統合設定]** フォーム（**[統合モジュール設定] > [HPOM]**）を開きます。
2. フォームの上端の **[統合を有効化]** チェックボックスをオフにし、フォームの下端の **[送信]** をクリックします。

変更はただちに有効になります。`ovjboss` を再起動する必要はありません。

必要な場合は、すべての NNMi 管理サーバーについてこのプロセスを繰り返します。

1 つの HPOM 管理サーバーについて

NNMi インシデントの 1 つの HPOM 管理サーバーへの転送のみを継続停止にするには、次のステップを実行します。

1 NNMi コンソールで、**[HP NNMi–HPOM 統合設定]** フォーム（**[統合モジュール設定] > [HPOM]**）を開きます。

2 **[追加の HPOM サーバー]** リストで、HPOM 管理サーバーを統合から切断するエントリ（1 つまたは複数）を削除するテキストを編集します。

[クリア] をクリックすると、リストからすべての HPOM サーバーが削除されます。

3 フォームの下端の **[送信]** をクリックします。

変更はただちに有効になります。`ovjboss` を再起動する必要はありません。

HP NNMi–HPOM 統合のトラブルシューティング

HPOM は転送されたインシデントを受信しない

統合が過去においては正常に動作していた場合は、構成の何か（たとえば、NNMi または HPOM のパスワード）が最近変更された可能性があります。NNMi–HPOM 統合設定の変更ページの 194 に説明してあるように統合設定を更新してから、この手順全体を段階的に行うこともできます。

1 NNMi コンソールで、**[HP NNMi–HPOM 統合設定]** フォーム（**[統合モジュール設定] > [HPOM]**）を開きます。

2 統合のステータスを確認するには、**[HP NNMi–HPOM 統合設定]** フォームの下端にある **[送信]** をクリックします（設定の変更はしません）。

新しいウィンドウにステータスメッセージが表示されます。

- HPOM サーバーへの接続に問題があることがメッセージに示されている場合、NNMi と HPOM は通信できません。この手順の 手順 3 を継続します。
- 正常な完了がメッセージに示されている場合は、NNMi が管理するデバイスからインシデントを受信するように HPOM が設定されていないことが問題であると考えられます。190 ページの手順 4（Windows 用の HPOM）と 192 ページの手順 4（UNIX 用の HPOM）に説明してあるように、HPOM で管理対象ノードとして設定されていない NNMi ソースノードから転送されたインシデントを HPOM は無視します。

3 次のように、HPOM 管理サーバーへの接続が正しく設定されていることを確認します。

a Web ブラウザで、次の URL を入力します。

https://*<omserver>*:*<port>*/opr-webservice//Incident.svc?wsdl

ここで、*<omserver>* と *<port>* はそれぞれ、**[HP NNMi–HPOM 統合設定]** フォームでの **[HPOM ホスト]** と **[HPOM ポート]** の値です。

b プロンプトが表示されたら、**[HP NNMi–HPOM 統合設定]** フォームから **[HPOM ユーザー]** の資格認定を入力します。

結果の Web ページは IWS を記述する XML ファイルです。

— XML ファイルが表示された場合、HPOM サーバーへの接続は正常に設定されています。手順 4 を継続します。

— エラーメッセージが表示された場合、HPOM 管理サーバーへの接続は正常に設定されていません。HPOM 管理者に連絡して、HPOM Web サービスに接続するために使用中の情報を確認してください。XML ファイルが表示されるまで、HPOM への接続のトラブルシューティングを継続します。

4 次のように、接続 NNMi が正常に設定されていることを確認します。

この手順の 手順 1 で NNMi コンソールに接続するために、このステップで説明してある情報を使用した場合は、NNMi コンソールに再接続する必要はありません。手順 5 を継続します。

a Web ブラウザで、次の URL を入力します。

http://*<NNMiserver>*:*<port>*/nnm/

ここで、*<NNMiserver>* と *<port>* はそれぞれ、**[HP NNMi–HPOM 統合設定]** フォームでの **[NNMi ホスト]** と **[NNMi ポート]** の値です。

次のコマンドによって返される NNMi 管理サーバーの完全修飾ドメイン名を使用します。

```
nnmofficialfqdn.ovpl -t
```

b プロンプトが表示されたら、管理者ロールで NNMi ユーザーの資格認定を入力します。

NNMi コンソールが表示されるはずです。NNMi コンソールが表示されない場合は NNMi 管理者に連絡して、NNMi に接続するために使用している情報を確認してください。NNMi コンソールが表示されるまで、NNMi への接続のトラブルシューティングを継続します。

Web サービスクライアントロールでユーザーとして NNMi コンソールにサインインすることはできません。

c **[NNMi ユーザー]** と **[NNMi パスワード]** の値を確認します。

— **[HP NNMi–HPOM 統合設定]** フォームにリストされている **[NNMi ユーザー]** が Administrator ロールを今も持っており、以前にこのユーザー名で NNMi コンソールに接続できた場合は、**[HP NNMi–HPOM 統合設定]** フォームに対応するパスワードを再入力します。

— **[HP NNMi–HPOM 統合設定]** フォームにリストされた **[NNMi ユーザー]** が Web Service Client ロールを持っている場合は、NNMi 管理者に連絡し、**[NNMi ユーザー]** と **[NNMi パスワード]** の値を確認してください。

パスワードは NNMi コンソールでは非表示です。NNMi ユーザー名のパスワードが何か確信がない場合は、NNMi 管理者に問い合わせするか、またはパスワードをリセットします。

5 この手順の 手順 3 と 手順 4 で使用して正常に接続できた値で、**[HP NNMi–HPOM 統合設定]** フォームを更新します。

詳細は、HP NNMi–HPOM Integration Configuration フォームリファレンスページの 198 を参照してください。

6 フォームの下端の **[送信]** をクリックします。

7 設定をテストするには、NNMi 管理サーバーでインシデントを生成し、それが HPOM 管理サーバーに到達するかどうか判定します。

転送されたインシデントの中に HPOM が受信しないものがある

HPOM ノードとインシデントフィルタを確認します。

HPOM 管理サーバーは、NNMi が管理するデバイスからのインシデントを受信するよう設定する必要があります。190 ページの手順 4（Windows 用の HPOM）と 192 ページの手順 4（UNIX 用の HPOM）に説明してあるように、HPOM で管理対象ノードとして設定されていない NNMi ソースノードから転送されたインシデントを HPOM は無視します。

NNMi ソースノードが HPOM 内で管理対象ノードとして設定されている場合は、**[HP NNMi–HPOM 統合設定]** フォームのインシデントフィルタ設定を確認します。フィルタをテストするには、NNMi 管理サーバーでインシデントを生成し、それが HPOM 管理サーバーに到達するかどうか判定します。

NNMi インシデント情報が HPOM メッセージブラウザでは入手できない

NNMi インシデントからの重要情報は、カスタムメッセージ属性として HPOM に渡されます。191 ページの手順 5（Windows 用の HPOM）と 192 ページの手順 5（UNIX 用の HPOM）に説明してあるように、NNMi インシデント用に 1 つまたは複数のカスタムメッセージ属性を追加します。

NNMi と HPOM が同期されない

どちらかの管理サーバーが到達不可能になった場合、NNMi インシデントビューのインシデントと HPOM アクティブメッセージブラウザのインシデントが一致しなくなる可能性があります。次のようにすると、NNMi HPOM 統合はインシデントを再同期できます。

- NNMi HPOM 統合モジュールが HPOM 管理サーバーを利用できなくなると、統合モジュールはその HPOM 管理サーバーの使用可能性を定期的に確認し、接続が再確立できた時点でインシデント転送を再開します。HPOM 管理サーバーへの接続が利用できるようになると、統合モジュールは、HPOM 管理サーバーの停止の間に失われた可能性のあるインシデントを転送します。
- 転送されたインシデントを HPOM ユーザーが肯定応答したとき、または肯定応答しなかったときに NNMi 管理サーバーが利用不能になった場合、NNMi は状態変化を受信しません。NNMi と HPOM はこのインシデントについて異なる状態を示す可能性があります。

統合がファイアウォールを経由して動作しない

NNMi 管理サーバーが、ホストとポートによって、HPOM IWS を直接取り扱えることを確認します。

HP NNMi–HPOM Integration Configuration フォームリファレンス

[HP NNMi–HPOM 統合設定] フォームには、NNMi と HPOM の間の通信用パラメータが含まれています。このフォームは、**[統合モジュール設定]** ワークスペースから利用できます。

Administrator ロールのある NNMi ユーザーのみが **[HP NNMi HPOM 統合設定]** フォームにアクセスできます。

[HP NNMi–HPOM 統合設定] フォームは、次の一般領域に関する情報を収集します。

- NNMi 管理サーバー接続
- HPOM Management Server Connection（HPOM 管理サーバー接続）
- 統合動作
- Incident Filter（インシデントフィルタ）

統合設定に変更を適用するには、**[HP NNMi–HPOM 統合設定]** フォームの値を更新し、**[送信]** をクリックします。

NNMi 管理サーバー接続

表 16 に、NNMi 管理サーバーへの接続パラメータをリストします。これは NNMi コンソールを開くために使用したのと同じ情報です。これらの値の多くを決定するには、NNMi コンソールセッションを起動する URL を調べます。NNMi 管理者と協力し、設定フォームのこのセクションに適切な値を決定します。

デフォルトの NNMi 設定は NNMi コンソールへの接続に http を使用します。http を使用するこの接続の設定の詳細は、https の有効化（対象：NNMi）ページの 73 を参照してください。

表 16　NNMi 管理サーバー接続情報

フィールド	説明
NNMi SSL Enabled（NNMi SSL 有効化）	接続プロトコル指定。 • NNMi コンソールが https を使用するよう設定されている場合は、**[NNMi SSL 有効化]** チェックボックスをオンにします。 • NNMi コンソールが http を使用するよう設定されている場合は、**[NNMi SSL 有効化]** チェックボックスをオフにします。これがデフォルト設定です。
NNMi ホスト	NNMi 管理サーバーの完全修飾ドメイン名。このフィールドには、NNMi コンソールへのアクセスに使用するホスト名があらかじめ入力されています。この値が、NNMi 管理サーバー上で **`nnmofficialfqdn.ovpl -t`** コマンド実行によって返された名前であることを確認します。

表 16　NNMi 管理サーバー接続情報（続く）

フィールド	説明
NNMi Port （NNMi ポート）	NNMi コンソールに接続するためのポート。このフィールドには、次のファイルで指定されているように、NNMi コンソールとの通信のために jboss アプリケーションサーバーが使用するポートがあらかじめ記入されています。 • *Windows*: `%NnmDataDir%\shared\nnm\conf\nnm.ports.properties` • *UNIX*: `$NnmDataDir/shared/nnm/conf/nnm.ports.properties` SSL 以外の接続では、`jboss.http.port` の値を使用します。これはデフォルトでは `80` または `8004` です（NNMi がインストールされたときに別の Web サーバーが存在するかどうかで、どちらかが決まります）。 SSL 接続には、`jboss.https.port` の値を使用します。これはデフォルトでは `443` です。
NNMi User （NNMi ユーザー）	NNMi コンソールに接続するためのユーザー名。このユーザーは、NNMi Administrator または Web Service Client のロールを持っている必要があります。
NNMi Password （NNMi パスワード）	指定の NNMi ユーザーのパスワード。

HPOM Management Server Connection（HPOM 管理サーバー接続）

表 17 に、HPOM 管理サーバー上で Web サーバーに接続するためのパラメータをリストします。HPOM 管理者と協力し、設定のこのセクションに適切な値を決定します。

表 17　HPOM 管理サーバー接続情報

HPOM サーバーパラメータ	説明
HPOM SSL Enabled （HPOM SSL 有効化）	接続プロトコル指定。 • HPOM が https を使用するよう設定されている場合は、**［HPOM SSL 有効化］** チェックボックスをオンにします。これがデフォルト設定です。 • HPOM が http を使用するよう設定されている場合は、**［HPOM SSL 有効化］** チェックボックスをオフにします。
HPOM Host （HPOM ホスト）	HPOM 管理サーバーの完全修飾ドメイン名。
HPOM Port （HPOM ポート）	HPOM Web サービスに接続するためのポート。どのポート番号を指定するか決定するには、HPOM 管理サーバーで以下を行います。 • *Windows* : IIS Manager のポート設定を調べます。これは **［スタート］** メニューから利用できます。たとえば、**［スタート］** > **［管理ツール］** > **［インターネット情報サービス (IIS) マネージャ］** です。 • *UNIX* : **`ovtomcatbctl -getconf`** コマンドを実行します。 このフィールドには `443` という値があらかじめ記入されています。Windows の場合の、HPOM への SSL 接続のデフォルトポートです。UNIX の場合、HPOM への SSL 接続のデフォルトポートは 8443 です。
HPOM User （HPOM ユーザー）	HPOM Administrator ロールのある、有効な HPOM ユーザーアカウント名。 *Windows* のみ : Windows オペレーティングシステムでは、HPOM は Microsoft インターネット情報サービス (IIS) を経由して動作し、ユーザー資格認定を認証します。Windows ユーザーは *`<Windows_domain>\<username>`* というフォーマットで指定します。
HPOM Password （HPOM パスワード）	指定の HPOM ユーザーのパスワード。

統合動作

表 18 に、統合動作を記述するパラメータをリストします。NNMi 管理者と協力し、設定のこのセクションに適切な値を決定します。

表 18　統合動作情報

フィールド	説明
Forward Only（転送のみ）	NNMi HPOM 統合モジュールの動作指定。デフォルトで、統合モジュールは、**[HP NNMi–HPOM Integration Configuration]** フォームで識別された HPOM 管理サーバーにインシデント肯定応答を転送し、同じサーバーからインシデント肯定応答を受信します。インシデント肯定応答の受信を無効にすることもできます。 • 単方向通信（HPOM にインシデントを転送するが、HPOM からのインシデント肯定応答は無視する）の場合は、**[転送のみ]** チェックボックスをオンにします。 • 双方向通信の場合は、**[転送のみ]** チェックボックスをオフにします。これがデフォルト動作です。
Holding period (minutes)（保持期間（分単位））	HPOM に設定済みインシデントを転送するまで待機できる時間（分単位）。この待機時間の間にインシデントが閉じられた場合（たとえば、SNMPLinkUp インシデントが SNMPLinkDown インシデントをキャンセルした場合）、HPOM はそのインシデントを受信することはありません。NNMi がただちにインシデントを転送するようにするには、値 0 を入力します。 デフォルト値は 5 分です。
Incident Filter（インシデントフィルタ）	インシデント転送を制限する NNMi インシデント属性を基礎にしたフィルタ。デフォルトのフィルタ (`nature=ROOTCAUSE origin=MANAGEMENTSOFTWARE`) は、NNMi が生成するすべての根本原因インシデントを指定します。フィルタを修正して、どのインシデントが HPOM に転送されるか変更できます。詳細は、Incident Filter（インシデントフィルタ）を参照してください。

Incident Filter（インシデントフィルタ）

インシデントフィルタは、**[インシデントフィルタ]** リスト内の全エントリの組み合わせです。同じ属性を持つフィルタエントリはフィルタを拡張します（論理和）。異なる属性を持つフィルタエントリはフィルタを制限します（論理積）。すべてのフィルタエントリがともに作用します。`(a AND b) OR c` というフォーマットのフィルタは作成できません。例として挙げたフィルタエントリは手順に従います。

インシデントフィルタを作成するには、以下のステップを行います。

1. NNMi コンソールで、**[HP NNMi–HPOM 統合設定]** フォーム（**[統合モジュール設定] > [HPOM]**）を開きます。
2. フィルタエントリを削除するには、**[インシデントフィルタ]** リストで、テキストを編集してエントリ（1 つまたは複数）を削除します。

[クリア] をクリックすると、リストからすべてのフィルタエントリが削除されます。

3. インシデントフィルタエントリを追加する方法
   a. **[名前]** リストから属性を選択します。サポートされている属性については、手順 c の表を参照してください。

b 実行する比較演算子を選択します。サポートされている演算子は次のとおりです。

— =

— !=

— <

— <=

— >

— >=

c 比較値を入力します。次表に、サポートされている属性、および各属性について受け入れられる値をリストします。

属性	可能な値
name（名前）	NNMi コンソールのインシデント設定を調べ、使用可能なインシデント名を決定します。
nature（性質）	• ROOTCAUSE • SECONDARYROOTCAUSE • SYMPTOM • SERVICEIMPACT • STREAMCORRELATION • NONE
origin（ソース）	• MANAGEMENTSOFTWARE • MANUALLYCREATED • SYMPTOM • REMOTELYGENERATED • SNMPTRAP • SYSLOG • OTHER
family（ファミリ）	• 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

属性	可能な値
category （カテゴリ）	• 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 （重要度）	• NORMAL • WARNING • MINOR • MAJOR • CRITICAL

4 すべてのフィルタエントリが定義されるまで、手順 3 を繰り返します。

5 フォームの下端の［**送信**］をクリックします。

インシデントフィルタの例

NNMi から HPOM に NodeDown インシデントを転送

```
name=NodeDown
```

NNMi から HPOM に NodeDown インシデントと InterfaceDown インシデントを転送

```
name=NodeDown
name=InterfaceDown
```

NNMi から HPOM に CiscoLinkDown インシデントを転送

```
name=CiscoLinkDown
```

severity（重要度）が MAJOR または MINOR の NNMi 管理イベントを転送

```
origin=Management Software
severity=MAJOR
severity=MINOR
```

severity（重要度）が少なくとも MINOR で nature（性質）が ROOTCAUSE または SERVICEIMPACT の NNMi インシデントを転送

```
severity>=MINOR
nature=ROOTCAUSE
nature=SERVICEIMPACT
```

インシデントフィルタの制限

すべてのフィルタエントリが結合して NNMi 管理サーバー用の 1 つのインシデントフィルタを作成するので、次の制限が適用されます。

- 記述された severity（重要度）がすべてのインシデントに適用されます。たとえば、severity（重要度）が MINOR 以上の NodeDown インシデントと severity（重要度）が MAJOR の InterfaceDown インシデントを転送するには、フィルタの severity（重要度）を `>=MINOR` に設定し、HPOM ロジックを使用して不要な InterfaceDown メッセージをフィルタで除外します。
- インシデントフィルタには、特定のソースノードへのインシデント転送を制限するメカニズムが備わっていません。HPOM 管理対象ノード（または外部ノード）設定は、HPOM が受信する転送済みインシデントを制限します。

HP NNMi–HPOM 統合ログファイルリファレンス

ログファイル

次のログファイルには、NNMi HPOM 統合用のメッセージが含まれています。

- NNMi 管理サーバー。
  - *Windows*:

    ```
    %NnmDataDir%\log\nnm\nnm-om.#.#.log
    %NnmDataDir%\log\nnm\jbossserver.#.#.log
    ```

  - *UNIX*:

    ```
    $NnmDataDir/log/nnm/nnm-om.#.#.log
    $NnmDataDir/log/nnm/jbossserver.#.#.log
    ```

NNMi 管理サーバーに関する情報に加えて、`nnm-om.#.#.log` ファイルには、HPOM 管理サーバーで発生したエラーに関するメッセージが含まれています。

- Windows 管理サーバー用の HPOM。

  ```
  %ovdatadir%\shared\server\log\om\incident-ws.trace.txt
  ```

- UNIX 管理サーバー用の HPOM。

  ```
  \var\opt\OV\log\om\incident_ws.0.en
  ```

たとえば、`nnm-om.#.#.log` ファイルのメッセージストリームの例については、ログファイルメッセージの例ページの 205 を参照してください。

nnm-om ログファイル内の詳細メッセージを有効にする

デフォルトでは、`nnm-om.#.#.log` ファイルは FINE ログレベルでメッセージを取得するよう設定されています。すべてのメッセージを取得できるようにするには、次のコマンドを実行します。

- *Windows*:

  ```
  %NnmInstallDir%\support\nnmsetlogginglevel.ovpl
  com.hp.ov.nnm.om FINEST
  ```

- *UNIX*:

  ```
  $NnmInstallDir/support/nnmsetlogginglevel.ovpl
  com.hp.ov.nnm.om FINEST
  ```

jboss-log4j ログファイルで統合通信メッセージを有効にする

デフォルトでは、jbossserver.#.#.log ファイルは、HP NNMiHPOM 統合モジュールからのメッセージを取得しません。NNMi と HPOM の間のすべての通信のログを有効にするには、次の手順を行います。

1 次のファイルをバックアップします。

- *Windows*:

  %NnmInstallDir%\nonOv\jboss\nms\server\nms\conf\jboss-log4j.xml

- *UNIX*:

  $NnmInstallDir/nonOv/jboss/nms/server/nms/conf/jboss-log4j.xml

2 テキストエディタで jboss-log4j.xml ファイルを開きます。

3 文字列 nnm-om を検索してから、この文字列に続く行（複数）をコメントアウトします。例：

```
<!-- Next 3 added due to nnm-om module -->
    <category name="org.apache.axis2">
        <priority value="WARN"/>
    </category>
    <category name="org.apache.commons.httpclient">
        <priority value="WARN"/>
    </category>
    <category name="httpclient.wire">
        <priority value="INFO"/>
    </category>
```

4 ovjboss を再起動します。

```
ovstop ovjboss
ovstart ovjboss
```

ログファイルメッセージの例

このセクションの例は、nnm-om.#.#.log ファイルにあります。以下の例があります。

- Good Start up Trace（良好なトレースの起動）
- UNIX 用に HPOM に正常に転送された NNMi インシデント
- UNIX 用に HPOM で正常に閉じられた NNMi インシデント
- 正しくない HPOM ログイン設定
- 正しくない NNMi ログイン設定
- Windows 用の HPOM からの肯定応答エラー
- UNIX 用の HPOM からの肯定応答エラー
- ノードが見つからないというエラー

Good Start up Trace（良好なトレースの起動）

```
Sep 2, 2008 10:07:00.921 AM [ThreadID:1] ALL: com.hp.ov.nms.admin.log.NnmOmFileHandler
init: Logging Handler initialized to log to file pattern: "C:/Documents and Settings/All
Users/Application Data/HP/HP BTO Software\log\nnm\nnm-om.%g.%u.log" up to a limit of 5
files of 1 MB each. Initially handling loggers:"com.hp.ov.nnm.om".

Sep 2, 2008 10:19:05.234 AM [ThreadID:61] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule init:
Initialized nnm: http://nnmserver.domain.com:80 with filter(/sys:onNotification/arg0
[(name='NodeDown' or name='InterfaceDown' or name='SNMPLinkDown')] )

Sep 2, 2008 10:19:05.265 AM [ThreadID:61] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule init:
Initialized om: https://omserver.domain.com:8444

Sep 2, 2008 10:19:05.281 AM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule$OmServer
setSubscription: OM unsubscribing from: omserver.domain.com

Sep 2, 2008 10:19:05.281 AM [ThreadID:61] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule disable:
NNM-OM module disabled

Sep 2, 2008 10:19:31.187 AM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule
subscribeNnm: NNM Subscription details: SubscriptionManager: http://NNMSERVER:80/
nms-sdk-notify/manage
SubscriptionID: urn:jbwse:58ea7aa4-ec0-4ec2-9949-895f8ca34f46
Expires: Tue Sep 02 10:29:16 BST 2008
Notify To: EPR {address=http://nnmserver.domain.com:80/NnmOmEventSinkBeanService/
NnmOmEventSinkBean, refParam=null}

Sep 2, 2008 10:19:31.218 AM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule$OmServer
setSubscription: OM unsubscribing from: omserver.domain.com

Sep 2, 2008 10:19:46.093 AM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.OmBean
subscribeIncidents: Listening to omserver.domain.com for incidents from:
nnmserver.domain.com

Sep 2, 2008 10:19:46.125 AM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule
subscribeOm: OM Subscription details: OM Subscription Keys:
   omserver.domain.com: f5433fc2-b55d-4334-b234-2861d6a65605

Sep 2, 2008 10:19:46.125 AM [ThreadID:61] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule enable:
NNM-OM module enabled

Sep 2, 2008 10:21:46.218 AM [ThreadID:63] FINE: com.hp.ov.nnm.om.im.NnmOmModule
doSweepForMissedIncidents: Sweeping for missed incidents for: omserver.domain.com

Sep 2, 2008 10:21:46.234 AM [ThreadID:63] FINE: com.hp.ov.nnm.om.im.NnmOmModule
doSweepQuery: Sweep query: select inc.uuid from Incident inc where inc.lifecycleState <>
4294967299 and (name='NodeDown' or name='InterfaceDown' or name='SNMPLinkDown') and
inc.id not in (select distinct i.id from Incident i, CustomIncidentAttribute cia where
cia.incident = i.id and cia.name like 'nnmOmEPR.omserver.domain.com')

Sep 2, 2008 10:22:00.890 AM [ThreadID:63] FINE: com.hp.ov.nnm.om.im.NnmOmModule
doSweepForMissedIncidents: Forwarding 1 incidents to: omserver.domain.com

Sep 2, 2008 10:22:44.453 AM [ThreadID:63] FINE: com.hp.ov.nnm.om.im.NnmOmModule
forwardNnmIncidentToOmServers: Forwarding to OM(omserver.domain.com), incident
08de1d45-5c57-4b61-a0d0-b9da119f066d

Sep 2, 2008 10:22:45.671 AM [ThreadID:63] FINE: com.hp.ov.nnm.om.im.NnmOmModule
forwardNnmIncidentToOmServers: Forwarded incident(08de1d45-5c57-4b61-a0d0-b9da119f066d)
response: https://omserver.domain.com:8444/opr-webservice/Incident.svc/
230e5b72-78cc-71dd-1875-1019f8490000

Sep 2, 2008 10:22:48.484 AM [ThreadID:67] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onOmNotification: onNotification(omserver.domain.com):
08de1d45-5c57-4b61-a0d0-b9da119f066d/open
```

UNIX 用に HPOM に正常に転送された NNMi インシデント

```
Sep 3, 2008 12:53:23.290 PM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onNewIncident: Forwarding incident(77f7bab0-7963-71dd-0142-ac1107ce0000) to:
omserver.domain.com:443

Sep 3, 2008 12:53:23.431 PM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onNewIncident: Forwarded incident(77f7bab0-7963-71dd-0142-ac1107ce0000) response: https://
omserver.domain.com/opr-webservice/Incident.svc/833cd1a2-630b-4ba1-956a-b5450bd15dce
```

UNIX 用に HPOM で正常に閉じられた NNMi インシデント

```
Sep 2, 2008 12:46:37.234 PM [ThreadID:91] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onOmNotification: onNotification(omserver.domain.com):
9a687f90-78e0-71dd-0713-1019f86f0000/closed

Sep 2, 2008 12:46:40.625 PM [ThreadID:91] FINE: com.hp.ov.nnm.om.im.NnmOmModule
synchronizeIncidentLifecycleState: Synchronizing NNM incident lifecycleState:
9a687f90-78e0-71dd-0713-1019f86f0000/com.hp.nms.incident.lifecycle.Closed
```

正しくない HPOM ログイン設定

```
Sep 2, 2008 10:54:46.500 AM [ThreadID:1] ALL: com.hp.ov.nms.admin.log.NnmOmFileHandler
init: Logging Handler initialized to log to file pattern: "C:/Documents and Settings/All
Users/Application Data/HP/HP BTO Software\log\nnm\nnm-om.%g.%u.log" up to a limit of 5
files of 1 MB each. Initially handling loggers:"com.hp.ov.nnm.om".

Sep 2, 2008 11:02:44.093 AM [ThreadID:61] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule init:
Initialized nnm: http://nnmserver.domain.com:80 with filter(/sys:onNotification/arg0
[(name='NodeDown' or name='InterfaceDown' or name='SNMPLinkDown')] )

Sep 2, 2008 11:02:44.109 AM [ThreadID:61] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule init:
Initialized om: https://omserver.domain.com:8444

Sep 2, 2008 11:02:44.140 AM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule$OmServer
setSubscription: OM unsubscribing from: omserver.domain.com

Sep 2, 2008 11:02:44.140 AM [ThreadID:61] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule disable:
NNM-OM module disabled

Sep 2, 2008 11:03:07.953 AM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule
subscribeNnm: NNM Subscription details: SubscriptionManager: http://NNMSERVER:80/
nms-sdk-notify/manage
SubscriptionID: urn:jbwse:d216f05e-caad-4b59-951b-89c57cb1d7b7
Expires: Tue Sep 02 11:12:57 BST 2008
Notify To: EPR {address=http://nnmserver.domain.com:80/NnmOmEventSinkBeanService/
NnmOmEventSinkBean, refParam=null}

Sep 2, 2008 11:03:07.968 AM [ThreadID:61] FINE: com.hp.ov.nnm.om.im.NnmOmModule$OmServer
setSubscription: OM unsubscribing from: omserver.domain.com

Sep 2, 2008 11:03:21.796 AM [ThreadID:61] SEVERE: com.hp.ov.nnm.om.im.NnmOmModule enable:
Error subscribing:  : org.apache.axis2.AxisFault: Transport error: 401 Error: No
AuthenticationProvider found for {0}
at org.apache.axis2.transport.http.HTTPSender.handleResponse(HTTPSender.java:298)
at org.apache.axis2.transport.http.HTTPSender.sendViaPost(HTTPSender.java:192)
at org.apache.axis2.transport.http.HTTPSender.send(HTTPSender.java:77)
at org.apache.axis2.transport.http.CommonsHTTPTransportSender.writeMessageWithCommons(
  CommonsHTTPTransportSender.java:327)
at org.apache.axis2.transport.http.CommonsHTTPTransportSender.invoke(
  CommonsHTTPTransportSender.java:206)
at org.apache.axis2.engine.AxisEngine.send(AxisEngine.java:396)
```

. . .

Sep 2, 2008 11:03:21.796 AM [ThreadID:61] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule enable: NNM-OM module enabled

Sep 2, 2008 11:04:26.406 AM [ThreadID:64] FINE: com.hp.ov.nnm.om.im.NnmOmModule$OmServer setSubscription: OM unsubscribing from: omserver.domain.com

Sep 2, 2008 11:04:36.328 AM [ThreadID:64] SEVERE: com.hp.ov.nnm.om.im.NnmOmModule checkOmSubscriptions: Failure subscribing to OM (omserver.domain.com): : org.apache.axis2.AxisFault: Transport error: 401 Error: No AuthenticationProvider found for {0}
at org.apache.axis2.transport.http.HTTPSender.handleResponse(HTTPSender.java:298)
at org.apache.axis2.transport.http.HTTPSender.sendViaPost(HTTPSender.java:192)
at org.apache.axis2.transport.http.HTTPSender.send(HTTPSender.java:77)
at org.apache.axis2.transport.http.CommonsHTTPTransportSender.writeMessageWithCommons(
  CommonsHTTPTransportSender.java:327)
at org.apache.axis2.transport.http.CommonsHTTPTransportSender.invoke(
  CommonsHTTPTransportSender.java:206)
at org.apache.axis2.engine.AxisEngine.send(AxisEngine.java:396)
. . .

正しくない NNMi ログイン設定

Sep 1, 2008 5:13:18.203 PM [ThreadID:1] ALL: com.hp.ov.nms.admin.log.NnmOmFileHandler init: Logging Handler initialized to log to file pattern: "C:/Documents and Settings/All Users/Application Data/HP/HP BTO Software\log\nnm\nnm-om.%g.%u.log" up to a limit of 5 files of 1 MB each. Initially handling loggers:"com.hp.ov.nnm.om".

Sep 1, 2008 5:21:40.343 PM [ThreadID:60] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule init: Initialized nnm: http://nnmserver.domain.com:80 with filter(/sys:onNotification/arg0 [(name='NodeDown' or name='InterfaceDown' or name='SNMPLinkDown')])

Sep 1, 2008 5:21:40.390 PM [ThreadID:60] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule init: Initialized om: http://omserver.domain.com:8081

Sep 1, 2008 5:21:40.390 PM [ThreadID:60] FINE: com.hp.ov.nnm.om.im.NnmOmModule$OmServer setSubscription: OM unsubscribing from: omserver.domain.com

Sep 1, 2008 5:21:40.390 PM [ThreadID:60] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule disable: NNM-OM module disabled

Sep 1, 2008 5:21:55.546 PM [ThreadID:60] SEVERE: com.hp.ov.nnm.om.im.NnmOmModule enable: Error subscribing: : javax.xml.ws.WebServiceException: java.io.IOException: Could not transmit message
at org.jboss.ws.core.jaxws.client.ClientImpl.handleRemoteException(ClientImpl.java:304)
at org.jboss.ws.core.jaxws.client.ClientImpl.invoke(ClientImpl.java:242)
at org.jboss.ws.core.jaxws.client.ClientProxy.invoke(ClientProxy.java:164)
at org.jboss.ws.core.jaxws.client.ClientProxy.invoke(ClientProxy.java:150)
at $Proxy569.subscribeOp(Unknown Source)
at com.hp.ov.nnm.om.im.NnmOmModule.doSubscribe(NnmOmModule.java:1129)
at com.hp.ov.nnm.om.im.NnmOmModule.subscribeNnm(NnmOmModule.java:922)
at com.hp.ov.nnm.om.im.NnmOmModule.enable(NnmOmModule.java:309)
at com.hp.ov.nnm.om.im.NnmOmModule$Enabler$EnableTask.run(NnmOmModule.java:1595)
at java.util.TimerThread.mainLoop(Timer.java:512)
at java.util.TimerThread.run(Timer.java:462)

Caused by: java.io.IOException: Could not transmit message
at org.jboss.ws.core.client.RemotingConnectionImpl.invoke(RemotingConnectionImpl.java:192)
at org.jboss.ws.core.client.SOAPRemotingConnection.invoke(SOAPRemotingConnection.java:77)
at org.jboss.ws.core.CommonClient.invoke(CommonClient.java:331)
at org.jboss.ws.core.jaxws.client.ClientImpl.invoke(ClientImpl.java:230)
. . .

Caused by: org.jboss.remoting.CannotConnectException: Can not connect http client invoker.

```
at org.jboss.remoting.transport.http.HTTPClientInvoker.useHttpURLConnection(
  HTTPClientInvoker.java:333)
at org.jboss.remoting.transport.http.HTTPClientInvoker.transport(
  HTTPClientInvoker.java:135)
at org.jboss.remoting.MicroRemoteClientInvoker.invoke(MicroRemoteClientInvoker.java:122)
at org.jboss.remoting.Client.invoke(Client.java:1634)
at org.jboss.remoting.Client.invoke(Client.java:548)
at org.jboss.ws.core.client.RemotingConnectionImpl.invoke(RemotingConnectionImpl.java:171)
. . .

Caused by: org.jboss.ws.WSException: Invalid HTTP server response [401] - Unauthorized
at org.jboss.ws.core.soap.SOAPMessageUnMarshaller.read(SOAPMessageUnMarshaller.java:72)
at org.jboss.remoting.transport.http.HTTPClientInvoker.readResponse(
  HTTPClientInvoker.java:473)
at org.jboss.remoting.transport.http.HTTPClientInvoker.useHttpURLConnection(
  HTTPClientInvoker.java:305)
. . .

Sep 1, 2008 5:21:55.546 PM [ThreadID:60] CONFIG: com.hp.ov.nnm.om.im.NnmOmModule enable:
NNM-OM module enabled
```

Windows 用の HPOM からの肯定応答エラー

次の SEVERE ログメッセージが発生するのは、NNMi 管理サーバーで既に閉じられている転送済み NNMi インシデントを HPOM オペレータが肯定応答するときです。このエラー状況は無視しても安全です。

```
Sep 2, 2008 12:46:37.234 PM [ThreadID:91] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onOmNotification: onNotification(omserver.domain.com):
9a687f90-78e0-71dd-0713-1019f86f0000/closed

Sep 2, 2008 12:46:40.625 PM [ThreadID:91] FINE: com.hp.ov.nnm.om.im.NnmOmModule
synchronizeIncidentLifecycleState: Synchronizing NNM incident lifecycleState:
9a687f90-78e0-71dd-0713-1019f86f0000/com.hp.nms.incident.lifecycle.Closed

Sep 2, 2008 12:46:45.984 PM [ThreadID:64] FINE: com.hp.ov.nnm.om.im.NnmOmEventSinkBean
onNotification: NnmOmEventSinkBean.onNotification: <id>450971566080</
id><uuid>9a687f90-78e0-71dd-0713-1019f86f0000</uuid><name>SNMPTest</
name><sourceUuid>null</sourceUuid><sourceName>none</sourceName><sourceNodeUuid>null</
sourceNodeUuid><sourceNodeName>16.26.120.61</
sourceNodeName><lifecycleState>com.hp.nms.incident.lifecycle.Closed</
lifecycleState><severity>MINOR</severity><priority>com.hp.nms.incident.priority.None</
priority><category>com.hp.nms.incident.category.Status</
category><family>com.hp.nms.incident.family.Node</family><nature>ROOTCAUSE</
nature><origin>SNMPTRAP</origin><duplicateCount>0</
duplicateCount><priority>com.hp.nms.incident.priority.None</priority><rcaActive>false</
rcaActive><formattedMessage><! [CDATA [Received SNMP test packet]] ></
formattedMessage><originOccurrenceTime>2008-09-02T12:16:36.000+01:00</
originOccurrenceTime><firstOccurrenceTime>2008-09-02T12:16:39.343+01:00</
firstOccurrenceTime><lastOccurrenceTime>2008-09-02T12:16:39.343+01:00</
lastOccurrenceTime><updateTime>2008-09-02T12:46:43.000+01:00</
updateTime><previousLifecycleState>com.hp.nms.incident.lifecycle.Registered</
previousLifecycleState><previousRcaActive>false</
previousRcaActive><cias><name>cia.snmpoid</name><type>STRING</
type><value>.1.3.6.1.4.1.999.1.0.1001</value></cias><cias><name>cia.address</
name><type>STRING</type><value>16.26.120.61</value></
cias><cias><name>nnmOmEPR.omserver.domain.com</name><type>STRING</type><value>https://
omserver.domain.com:8444/opr-webservice/Incident.svc/
70f54e8c-78df-71dd-1875-1019f8490000</value></cias><cias><name>.1.3</
name><type>ASN_INTEGER</type><value>100</value></cias>

Sep 2, 2008 12:46:46.000 PM [ThreadID:64] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onIncidentUpdate: Incident update received: 9a687f90-78e0-71dd-0713-1019f86f0000
<id>450971566080</id><uuid>9a687f90-78e0-71dd-0713-1019f86f0000</uuid><name>SNMPTest</
name><sourceUuid>null</sourceUuid><sourceName>none</sourceName><sourceNodeUuid>null</
```

sourceNodeUuid><sourceNodeName>16.26.120.61</
sourceNodeName><lifecycleState>com.hp.nms.incident.lifecycle.Closed</
lifecycleState><severity>MINOR</severity><priority>com.hp.nms.incident.priority.None</
priority><category>com.hp.nms.incident.category.Status</
category><family>com.hp.nms.incident.family.Node</family><nature>ROOTCAUSE</
nature><origin>SNMPTRAP</origin><duplicateCount>0</
duplicateCount><priority>com.hp.nms.incident.priority.None</priority><rcaActive>false</
rcaActive><formattedMessage><! [CDATA [Received SNMP test packet]] ></
formattedMessage><originOccurrenceTime>2008-09-02T12:16:36.000+01:00</
originOccurrenceTime><firstOccurrenceTime>2008-09-02T12:16:39.343+01:00</
firstOccurrenceTime><lastOccurrenceTime>2008-09-02T12:16:39.343+01:00</
lastOccurrenceTime><updateTime>2008-09-02T12:46:43.000+01:00</
updateTime><previousLifecycleState>com.hp.nms.incident.lifecycle.Registered</
previousLifecycleState><previousRcaActive>false</
previousRcaActive><cias><name>cia.snmpoid</name><type>STRING</
type><value>.1.3.6.1.4.1.999.1.0.1001</value></cias><cias><name>cia.address</
name><type>STRING</type><value>16.26.120.61</value></
cias><cias><name>nnmOmEPR.omserver.domain.com</name><type>STRING</type><value>https://
omserver.domain.com:8444/opr-webservice/Incident.svc/
70f54e8c-78df-71dd-1875-1019f8490000</value></cias><cias><name>.1.3</
name><type>ASN_INTEGER</type><value>100</value></cias>

```
Sep 2, 2008 12:46:46.031 PM [ThreadID:64] FINE: com.hp.ov.nnm.om.im.NnmOmModule
ackNnmIncidentToOm: Acknowledging incident 9a687f90-78e0-71dd-0713-1019f86f0000
omIncident(70f54e8c-78df-71dd-1875-1019f8490000) to: omserver.domain.com
```

Sep 2, 2008 12:46:47.078 PM [ThreadID:64] SEVERE: com.hp.ov.nnm.om.im.OmBean acknowledgeIncident: Om acknowledge failure: : org.apache.axis2.AxisFault: The service cannot comply with the request due to internal processing errors.

```
at org.apache.axis2.util.Utils.getInboundFaultFromMessageContext(Utils.java:486)
at
org.apache.axis2.description.OutInAxisOperationClient.handleResponse(OutInAxisOperation.ja
va:343)
at org.apache.axis2.description.OutInAxisOperationClient.send(OutInAxisOperation.java:389)
at
org.apache.axis2.description.OutInAxisOperationClient.executeImpl(OutInAxisOperation.java:
211)
at org.apache.axis2.client.OperationClient.execute(OperationClient.java:163)
at com.hp.ov.opr.ws.client.axis.IncidentServiceStub.Close(IncidentServiceStub.java:756)
at com.hp.ov.opr.ws.client.axis.OmWsClient.closeOp(OmWsClient.java:411)
at com.hp.ov.nnm.om.im.OmBean.acknowledgeIncident(OmBean.java:194)
. . .
```

UNIX 用の HPOM からの肯定応答エラー

次の SEVERE ログメッセージが発生するのは、NNMi 管理サーバーで既に閉じられている転送済み NNMi インシデントを HPOM オペレータが肯定応答するときです。このエラー状況は無視しても安全です。

```
Sep 3, 2008 12:53:23.728 PM [ThreadID:119] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onOmNotification: onNotification(omserver.domain.com):
77f7bab0-7963-71dd-0142-ac1107ce0000/open

Sep 3, 2008 12:54:22.307 PM [ThreadID:121] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onOmNotification: onNotification(omserver.domain.com):
77f7bab0-7963-71dd-0142-ac1107ce0000/closed

Sep 3, 2008 12:54:22.589 PM [ThreadID:121] FINE: com.hp.ov.nnm.om.im.NnmOmModule
synchronizeIncidentLifecycleState: Synchronizing NNM incident lifecycleState:
77f7bab0-7963-71dd-0142-ac1107ce0000/com.hp.nms.incident.lifecycle.Closed

Sep 3, 2008 12:54:23.089 PM [ThreadID:120] FINE: com.hp.ov.nnm.om.im.NnmOmEventSinkBean
onNotification: NnmOmEventSinkBean.onNotification: <id>1404454305792</
id><uuid>77f7bab0-7963-71dd-0142-ac1107ce0000</uuid><name>SNMPColdStart</
name><sourceUuid>null</sourceUuid><sourceName>none</
```

sourceName><sourceNodeUuid>3d52aff2-eb9d-45de-8e66-3b882fc6e533</
sourceNodeUuid><sourceNodeName>chinnm72</
sourceNodeName><lifecycleState>com.hp.nms.incident.lifecycle.Closed</
lifecycleState><severity>NORMAL</severity><priority>com.hp.nms.incident.priority.None</
priority><category>com.hp.nms.incident.category.Fault</
category><family>com.hp.nms.incident.family.Node</family><nature>SYMPTOM</
nature><origin>SNMPTRAP</origin><duplicateCount>0</
duplicateCount><priority>com.hp.nms.incident.priority.None</priority><rcaActive>false</
rcaActive><formattedMessage><! [CDATA [Agent Up with Possible Changes (coldStart Trap)]]
></formattedMessage><originOccurrenceTime>2008-09-03T12:53:21.000+10:00</
originOccurrenceTime><firstOccurrenceTime>2008-09-03T12:53:21.259+10:00</
firstOccurrenceTime><lastOccurrenceTime>2008-09-03T12:53:21.259+10:00</
lastOccurrenceTime><updateTime>2008-09-03T12:54:22.964+10:00</
updateTime><previousLifecycleState>com.hp.nms.incident.lifecycle.Registered</
previousLifecycleState><previousRcaActive>false</
previousRcaActive><cias><name>cia.snmpoid</name><type>STRING</
type><value>.1.3.6.1.6.3.1.1.5.1</value></cias><cias><name>cia.address</
name><type>STRING</type><value>172.17.7.206</value></
cias><cias><name>nnmOmEPR.omserver.domain.com</name><type>STRING</type><value>https://
chiomw71.simnetad.simplot.com.au/opr-webservice/Incident.svc/
833cd1a2-630b-4ba1-956a-b5450bd15dce</value></cias>

Sep 3, 2008 12:54:23.089 PM [ThreadID:120] FINE: com.hp.ov.nnm.om.im.NnmOmModule
onIncidentUpdate: Incident update received: 77f7bab0-7963-71dd-0142-ac1107ce0000
<id>1404454305792</id><uuid>77f7bab0-7963-71dd-0142-ac1107ce0000</
uuid><name>SNMPColdStart</name><sourceUuid>null</sourceUuid><sourceName>none</
sourceName><sourceNodeUuid>3d52aff2-eb9d-45de-8e66-3b882fc6e533</
sourceNodeUuid><sourceNodeName>chinnm72</
sourceNodeName><lifecycleState>com.hp.nms.incident.lifecycle.Closed</
lifecycleState><severity>NORMAL</severity><priority>com.hp.nms.incident.priority.None</
priority><category>com.hp.nms.incident.category.Fault</
category><family>com.hp.nms.incident.family.Node</family><nature>SYMPTOM</
nature><origin>SNMPTRAP</origin><duplicateCount>0</
duplicateCount><priority>com.hp.nms.incident.priority.None</priority><rcaActive>false</
rcaActive><formattedMessage><! [CDATA [Agent Up with Possible Changes (coldStart Trap)]]
></formattedMessage><originOccurrenceTime>2008-09-03T12:53:21.000+10:00</
originOccurrenceTime><firstOccurrenceTime>2008-09-03T12:53:21.259+10:00</
firstOccurrenceTime><lastOccurrenceTime>2008-09-03T12:53:21.259+10:00</
lastOccurrenceTime><updateTime>2008-09-03T12:54:22.964+10:00</
updateTime><previousLifecycleState>com.hp.nms.incident.lifecycle.Registered</
previousLifecycleState><previousRcaActive>false</
previousRcaActive><cias><name>cia.snmpoid</name><type>STRING</
type><value>.1.3.6.1.6.3.1.1.5.1</value></cias><cias><name>cia.address</
name><type>STRING</type><value>172.17.7.206</value></
cias><cias><name>nnmOmEPR.omserver.domain.com</name><type>STRING</type><value>https://
chiomw71.simnetad.simplot.com.au/opr-webservice/Incident.svc/
833cd1a2-630b-4ba1-956a-b5450bd15dce</value></cias>

Sep 3, 2008 12:54:23.104 PM [ThreadID:120] FINE: com.hp.ov.nnm.om.im.NnmOmModule
ackNnmIncidentToOm: Acknowledging incident 77f7bab0-7963-71dd-0142-ac1107ce0000
omIncident(833cd1a2-630b-4ba1-956a-b5450bd15dce) to: omserver.domain.com

Sep 3, 2008 12:54:23.229 PM [ThreadID:120] SEVERE: com.hp.ov.nnm.om.im.OmBean acknowledgeIncident: Om acknowledge failure: : org.apache.axis2.AxisFault: The destination represented by the WS-Addressing To EPR with address Unable to find specified incident 833cd1a2-630b-4ba1-956a-b5450bd15dce
at omwWS.DataAccess.WmiProxy.executeMassCommand(Command cmd, List`1 incidentIdList) in
S:\DE\oprweb\om-ws\server\omw\Source\Resource Access\omwWS.DataAccess\WmiProxy.cs:line 576
at omwWS.ServiceImplementation.Incident.Close(CloseRequest request) in
S:\DE\oprweb\om-ws\server\omw\Source\Service
Interface\omwWS.ServiceImplementation\Incident.cs:line 1080 cannot be reached.
at
org.apache.axis2.handlers.addressing.AddressingInFaultHandler.invoke(AddressingInFaultHand
ler.java:114)
at org.apache.axis2.engine.Phase.invoke(Phase.java:292)
at org.apache.axis2.engine.AxisEngine.invoke(AxisEngine.java:212)
at org.apache.axis2.engine.AxisEngine.receive(AxisEngine.java:132)

```
at
org.apache.axis2.description.OutInAxisOperationClient.handleResponse(OutInAxisOperation.ja
va:336)
at org.apache.axis2.description.OutInAxisOperationClient.send(OutInAxisOperation.java:389)
at
org.apache.axis2.description.OutInAxisOperationClient.executeImpl(OutInAxisOperation.java:
211)
at org.apache.axis2.client.OperationClient.execute(OperationClient.java:163)
at com.hp.ov.opr.ws.client.axis.IncidentServiceStub.Close(IncidentServiceStub.java:756)
at com.hp.ov.opr.ws.client.axis.OmWsClient.closeOp(OmWsClient.java:411)
at com.hp.ov.nnm.om.im.OmBean.acknowledgeIncident(OmBean.java:194)
. . .
```

ノードが見つからないというエラー

ソースが HPOM 管理対象ノードとして正しく設定されていないため、HPOM が転送済み NNMi インシデントのソースを認識しません。

```
Sep 9, 2008 8:40:51.796 AM [ThreadID:75] SEVERE: com.hp.ov.nnm.om.im.NnmOmModule getNode:
Error retrieving node: null : com.hp.ov.nms.sdk.node.NmsNodeException: Cannot interrogate
model:
at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:3
9)
at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImp
l.java:27)
at java.lang.reflect.Constructor.newInstance(Constructor.java:494)
at org.jboss.ws.metadata.umdm.FaultMetaData.toServiceException(FaultMetaData.java:384)
at
org.jboss.ws.core.jaxws.SOAPFaultHelperJAXWS.getSOAPFaultException(SOAPFaultHelperJAXWS.ja
va:129)
at
org.jboss.ws.core.jaxws.binding.SOAP11BindingJAXWS.throwFaultException(SOAP11BindingJAXWS.
java:109)
at org.jboss.ws.core.CommonSOAPBinding.unbindResponseMessage(CommonSOAPBinding.java:567)
at org.jboss.ws.core.CommonClient.invoke(CommonClient.java:365)
at org.jboss.ws.core.jaxws.client.ClientImpl.invoke(ClientImpl.java:230)
at org.jboss.ws.core.jaxws.client.ClientProxy.invoke(ClientProxy.java:164)
at org.jboss.ws.core.jaxws.client.ClientProxy.invoke(ClientProxy.java:150)
at $Proxy583.getNodes(Unknown Source)
. . .
```

HP Universal Configuration Management Database（HP ユニバーサル設定管理データベース）

HP Universal Configuration Management Database（UCMDB、ユニバーサル設定管理データベース）は、HP Discovery and Dependency mapping（DDM、検出および依存関係マッピング）へのネイティブ統合によって、インフラストラクチャとアプリケーションの関係についての最新で正確な情報を自動的に維持します。UCMDB は以下のタスクに有益です。

- 影響モデル化を使用し、変更が行われる前に、インフラストラクチャとアプリケーションに対する変更の徐々に進行する効果を示します。
- 検出された変更履歴によって、実際に計画済みの変更または未計画の変更を追跡します。
- 既存のデータリポジトリの認識によって、環境の信頼できる共有ビューを得ます。

UCMDB の購入については、HP 販売員にお尋ねください。

この章では、以下の内容を記載しています。

- HP NNMi–HP UCMDB 統合
- HP NNMi–HP UCMDB 統合の有効化
- HP NNMi–HP UCMDB 統合の使用法
- HP NNMi–HP UCMDB 統合設定の変更
- HP NNMi–HP UCMDB 統合の無効化
- HP NNMi–HP UCMDB 統合のトラブルシューティング
- HP NNMi–HP UCMDB Integration Configuration フォームリファレンス

HP NNMi–HP UCMDB 統合

HP NNMi–HP UCMDB 統合は UCMDB と NNMi トポロジ情報を共有します。UCMDB は、設定項目 (CI) として NNMi トポロジに各デバイスを保存します。UCMDB は、Discovery and Dependency Mapping（DDM、検出と依存関係マッピング）パターンを NNMi トポロジ用の CI に適用し、デバイス障害の影響を予測します。この影響分析は、UCMDB ユーザーインタフェースおよび NNMi コンソールから入手できます。

値

HP NNMi HP UCMDB 統合によって、NNMi はネットワークデバイス関係の信頼できるソースになることができます。

サポートされるバージョン

この章の情報は、以下の製品バージョンに当てはまります。

- UCMDB バージョン 8.00 以降
- NNMi バージョン 8.10 以降

NNMi と UCMDB は同じコンピュータにインストールできません。これら 2 つの製品は、以下の構成のどちらかである異なるコンピュータにインストールする必要があります。

- 異なるオペレーティングシステム。たとえば、NNMi 管理サーバーは Linux システムであり、UCMDB サーバーは Windows システムです。
- 同じオペレーティングシステム。たとえば、NNMi 管理サーバーは Windows システムであり、UCMDB サーバーは 2 番目の Windows システムです。

サポートされているハードウェアプラットフォームおよびオペレーティングシステムの最新情報については、両方の製品の対応マトリックスを参照してください。

ドキュメント

この章では、UCMDB と通信するように NNMi を設定する方法、および NNMi コンソールの統合を使用する方法について説明します。

*HP Universal CMDB–HP Network Node Manager (NNMi) 統合ガイド*は、UCMDB 製品メディアに含まれており、NNMi と通信するために UCMDB を設定する方法について説明しています。UCMDB ユーザーインタフェースから統合を使用する方法についても説明します。

HP NNMi–HP UCMDB 統合の有効化

1. 次のように、UCMDB サーバーで、UCMDB と NNMi の間の接続を設定し、統合をカスタマイズします。
   a. UCMDB–NNMi 統合検出パッケージを導入します。
   b. NNMi トポロジデータ用に DDM プローブを設定および起動します。
   c. Network NNM レイヤー 2 ジョブを設定します。

   詳細は、*HP Universal CMDB–HP Network Node Manager (NNMi) 統合ガイド*を参照してください。
2. 次のように、NNMi 管理サーバーで、NNMi と UCMDB の間の接続を設定します。
   a. NNMi コンソールで、**[HP NNMi–HP UCMDB 統合設定]** フォームを開きます（**[統合モジュール設定]** > **[HP UCMDB]**）。
   b. **[統合を有効化]** チェックボックスをオンにし、フォームの残りのフィールドをアクティブにします。

c NNMi 管理サーバーへの接続情報を入力します。

これらのフィールドの詳細は、NNMi 管理サーバー接続ページの 218 を参照してください。

d UCMDB サーバーへの接続情報を入力します。

これらのフィールドの詳細は、UCMDB サーバー接続ページの 219 を参照してください。

e フォームの下端の［**送信**］をクリックします。

新しいウィンドウにステータスメッセージが表示されます。UCMDB サーバーへの接続に問題があることをメッセージが示している場合は、［**HP NNMi–HP UCMDB 統合設定**］フォームをふたたび開き（またはメッセージウィンドウで［**ALT**］+［**LEFT ARROW**］を押し）、エラーメッセージのテキストが指示するように、UCMDB サーバーへの接続の値を調整します。

3 NNMi 管理サーバーで、次のように、統合をカスタマイズします。

a NNMi コンソールで、［**HP NNMi–HP UCMDB 統合設定**］フォームを開きます（［**統合モジュール設定**］>［**HP UCMDB**］）。

b 以下のフィールドに値を入力します。

— ［**HP UCMDB 相互関係ルールプレフィックス**］

— ［**HP UCMDB 影響重要度レベル (1-9)**］

これらのフィールドの詳細は、統合動作ページの 219 を参照してください。

c フォームの下端の［**送信**］をクリックします。

HP NNMi–HP UCMDB 統合の使用法

次のように、HP NNMi–HP UCMDB 統合を有効にすると、次の 2 つの URL アクションが NNMi コンソールに追加されます。

- ［**UCMDB 影響対象 CI の検出**］アクション。説明は影響対象 CI の表示にあります。
- ［**UCMDB の CI を開く**］アクション。説明は UCMDB CI の表示にあります。

UCMDB ユーザーインタフェースから統合を使用する場合の詳細は、*HP Universal CMDB–HP Network Node Manager (NNMi) 統合ガイド*を参照してください。

影響対象 CI の表示

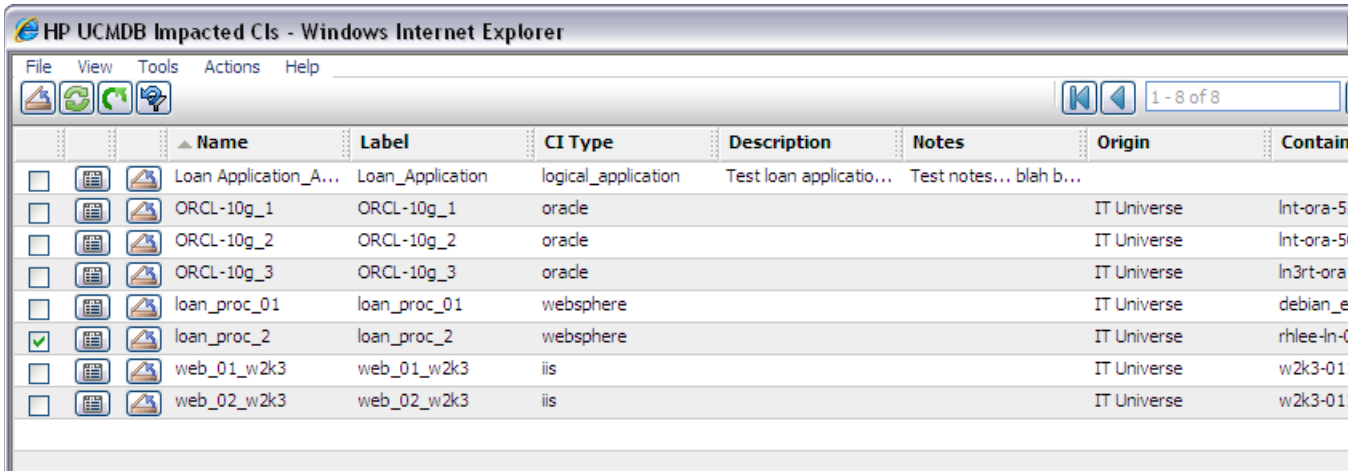
［**UCMDB 影響対象 CI の検出**］アクションは、選択したノードまたはインタフェースに関して（相互関係ルール、および統合用に設定された重要度レベルに従って）影響を受ける UCMDB 設定項目のリストを表示します。このアクションは次の NNMi コンソールの場所から入手できます。

- 任意のノードインベントリビュー
- 任意のインタフェースインベントリビュー
- 任意のマップビュー（選択したノードまたはインタフェース付き）
- 任意のインシデントブラウザ

[**UCMDB 影響対象 CI の検出**] アクションは、NNMi トポロジ内のすべてのノードおよびインタフェースについて利用できます。これらのオブジェクトが UCMDB データベースでモデル化されているかどうかは無関係です。

図 14 に、[**UCMDB 影響対象 CI の検出**] アクション例の結果を示します。

図 14　影響対象 CI テーブルの例



UCMDB CI の表示

[**UCMDB の CI を開く**] アクションは、選択した CI に関する UCMDB 情報を表示します。

HP NNMi–HP UCMDB 統合設定の変更

1. NNMi コンソールで、[**HP NNMi–HP UCMDB 統合設定**] フォームを開きます（[**統合モジュール設定**] > [**HP UCMDB**]）。
2. 該当するように値を変更します。

   このフォームのフィールドの詳細は、HP NNMi–HP UCMDB Integration Configuration フォームリファレンスページの 217 を参照してください。
3. フォームの上端の[**統合を有効化**]チェックボックスがオンであることを確認し、フォームの下端の [**送信**] をクリックします。

変更はただちに有効になります。ovjboss を再起動する必要はありません。

HP NNMi–HP UCMDB 統合の無効化

1 NNMi コンソールで、**[HP NNMi–HP UCMDB 統合設定]** フォームを開きます（**[統合モジュール設定]** > **[HP UCMDB]**）。

2 フォームの上端の **[統合を有効化]** チェックボックスをオフにし、フォームの下端の **[送信]** をクリックします。

統合 URL アクションはもう使用できません。

変更はただちに有効になります。`ovjboss` を再起動する必要はありません。

HP NNMi–HP UCMDB 統合のトラブルシューティング

トラブルシューティング情報は、利用できるようになったときに追加されます。

ドキュメントの更新バージョン取得の詳細は、使用可能な製品ドキュメントページの 3を参照してください。

HP NNMi–HP UCMDB Integration Configuration フォームリファレンス

[HP NNMi–HP UCMDB 統合設定] フォームには、NNMi と HPOM の間の通信用パラメータが含まれています。このフォームは、**[統合モジュール設定]** ワークスペースから利用できます。

Administrator ロールのある NNMi ユーザーのみが **[HP NNMi–HP UCMDB 統合設定]** フォームにアクセスできます。

[HP NNMi–HP UCMDB 統合設定] フォームは、以下の一般領域に関する情報を収集します。

- NNMi 管理サーバー接続
- UCMDB サーバー接続
- 統合動作

統合設定に変更を適用するには、**[HP NNMi–HP UCMDB 統合設定]** フォームの値を更新し、**[送信]** をクリックします。

NNMi 管理サーバー接続

表 19 に、NNMi 管理サーバーへの接続パラメータをリストします。これは NNMi コンソールを開くために使用したのと同じ情報です。これらの値の多くを決定するには、NNMi コンソールセッションを起動する URL を調べます。NNMi 管理者と協力し、設定フォームのこのセクションに適切な値を決定します。

デフォルトの NNMi 設定は NNMi コンソールへの接続に http を使用します。http を使用するこの接続の設定の詳細は、https の有効化（対象：NNMi）ページの 73 を参照してください。

表 19　NNMi 管理サーバー情報

フィールド	説明
NNMi SSL 有効化	接続プロトコル指定。 • NNMi コンソールが https を使用するよう設定されている場合は、**[NNMi SSL 有効化]** チェックボックスをオンにします。 • NNMi コンソールが http を使用するよう設定されている場合は、**[NNMi SSL 有効化]** チェックボックスをオフにします。これがデフォルト設定です。
NNMi ホスト	NNMi 管理サーバーの完全修飾ドメイン名。このフィールドには、NNMi コンソールへのアクセスに使用するホスト名があらかじめ入力されています。この値が、NNMi 管理サーバー上で **`nnmofficialfqdn.ovpl -t`** コマンド実行によって返された名前であることを確認します。
NNMi Port （NNMi ポート）	NNMi コンソールに接続するためのポート。このフィールドには、次のファイルで指定されているように、NNMi コンソールとの通信のために jboss アプリケーションサーバーが使用するポートがあらかじめ記入されています。 • *Windows*: `%NnmDataDir%\shared\nnm\conf\nnm.ports.properties` • *UNIX*: `$NnmDataDir/shared/nnm/conf/nnm.ports.properties` SSL 以外の接続では、`jboss.http.port` の値を使用します。これはデフォルトでは `80` または `8004` です（NNMi がインストールされたときに別の Web サーバーが存在するかどうかで、どちらかが決まります）。 SSL 接続には、`jboss.https.port` の値を使用します。これはデフォルトでは `443` です。
NNMi User （NNMi ユーザー）	NNMi コンソールに接続するためのユーザー名。このユーザーは、NNMi Administrator または Web Service Client のロールを持っている必要があります。
NNMi Password （NNMi パスワード）	指定の NNMi ユーザーのパスワード。

UCMDB サーバー接続

表 20 に、UCMDB サーバー上で Web サーバーに接続するためのパラメータをリストします。UCMDB 管理者と協力し、設定のこのセクションに適切な値を決定します。

表 20　UCMDB 管理サーバー情報

HPOM サーバーパラメータ	説明
HP UCMDB SSL Enabled（HP UCMDB SSL 有効化）	UCMDB Web サービスに接続するための接続プロトコル指定。 • UCMDB Web サービスが https を使用するよう設定されている場合は、**[HP UCMDB SSL 有効化]** チェックボックスをオンにします。これがデフォルト設定です。 • UCMDB Web サービスが http を使用するよう設定されている場合は、**[HP UCMDB SSL 有効化]** チェックボックスをオフにします。
HP UCMDB Host（HP UCMDB ホスト）	UCMDB サーバーの完全修飾ドメイン名。
HP UCMDB Port（HP UCMDB ポート）	UCMDB Web サービスに接続するためのポート。 デフォルトの UCMDB 設定を使用する場合は、ポート 8080 を使用します（UCMDB への SSL 接続の場合）。
HP UCMDB Port（HP UCMDB ユーザー）	UCMDB Administrator ロールのある、有効な UCMDB ユーザーアカウント名。
HP UCMDB Port（HP UCMDB パスワード）	指定の UCMDB ユーザーのパスワード。

統合動作

表 21 に、統合動作を記述するパラメータをリストします。UCMDB 管理者と協力し、設定のこのセクションに適切な値を決定します。

表 21　統合動作情報

フィールド	説明
[HP UCMDB 相互関係ルールプレフィックス]	NNMi トポロジオブジェクトに適用される UCMDB 影響相互関係ルールのプレフィックス。
[HP UCMDB 影響重要度レベル (1-9)]	UCMDB 影響相互関係ルールを適用する重要度レベル。HP は、最高の重要度 9 を使用して、可能な影響の計算にすべてのルールを組み込むことをお勧めします。

AlarmPoint

AlarmPoint は、対話型のアラート生成アプリケーションで、重要なイベントを収集し、内容を補強してから、各種の通信デバイス上の適切なユーザーに転送し、そのユーザーが問題点の解決、エスカレート、および他のユーザーの支援を得られることなどが可能なように設計されています。

AlarmPoint システムは、自動化エンジンまたは HP Network Node Manager i Software などのインテリジェントアプリケーションの音声関連のインタフェースとなるように、統合できます。NNMi と統合すると、注意が必要なイベントを検出したときに、AlarmPoint から、適切な担当者、ベンダー、カスタマに対して、電話をかける、またはポケベル、インスタントメッセージ、電子メールなどを送信できるようになります。

また、AlarmPoint システムでは、複数のデバイスを通じてエスカレートし、いずれかの担当者が解決を引き受けるか、または解決するまで、その処理を継続します。AlarmPoint システムでは、通知先のユーザーに対して、NNMi との双方向の通信チャネルを提供します。応答は NNMi 管理サーバーで即時に処理されるため、イベントのリモート解決が実現できます。

AlarmPoint のご購入については、当社営業担当にお尋ねください。

この章では、以下の内容を記載しています。

- HP NNMi–AlarmPoint 統合
- HP NNMi–AlarmPoint 統合を有効にする
- HP NNMi–AlarmPoint 統合の使用法
- HP NNMi–AlarmPoint 統合を無効にする
- HP NNMi–AlarmPoint 統合のトラブルシューティング

HP NNMi–AlarmPoint 統合

NNMi を使ってネットワークデバイスの監視と管理を担当するネットワーク管理者は、HP Network Node Manager i-series Software 環境に AlarmPoint を統合することで、インテリジェントなアラート生成手段、および個人使用の通信ツールと NNMi 間の双方向通信メカニズムを用意できます。

HP NNMi–AlarmPoint 統合は、NNMi のインシデントタイプの設定を通じて、NNMi から AlarmPoint へのイベント通知をサポートします。また、原因となったインシデントの受諾、その優先度の変更、注釈の追加など、AlarmPoint から NNMi へのインバウンドアクションもサポートします。

価値

HP NNMi–AlarmPoint 統合を使うと、適切な技術者に対して、音声、電子メール、ポケベル、その他のデバイスを通じて、直接通知できます。イベント解決の担当者は、障害に関する情報を受信し、イベントの受諾、無視、注釈追加、優先度変更などをリアルタイムで判断できます。

受信者がリモートデバイスからの応答を選択した後、AlarmPoint は NNMi インシデントをリアルタイムで更新します。このようなメカニズムの利点は、処理が即座に行われることです。すなわち、運用スタッフが障害または誤動作に気付いて、適切な担当者を判断し、その担当者に手動で通知するよりも、はるかに迅速に行われることです。単純なアクションで各種デバイスのインシデントを更新できるため、イベント解決の担当者にとっては、多くの問題を処理したり、インシデントの現在のステータスを他のチームメンバーに通知するための高速な手段となります。

処理中には、AlarmPoint は、すべての通知、応答、アクションをログに記録します。また、AlarmPoint は、元の NNMi インシデントに対してステータス情報を注釈として自動的に付加します。

AlarmPoint 製品には、各ジョブに適切な担当者を割り当てるための、セルフサービスの Web ベースユーザーインタフェース機能が備わっています。また、AlarmPoint には、オプションで、機能が強化されたサブスクリプションパネルも用意されており、これを使うと、NNMi インシデントに対して、サブスクリプションするかを判断したり、または自動的にサブスクリプションさせることが可能になります。

AlarmPoint の機能限定版である AlarmPoint Express は、NNMi の特定バージョンで使用できます。AlarmPoint Express for HP NNMi は、限定された機能セットで、NNMi 統合が設定済みです。AlarmPoint Express は、AlarmPoint 製品ファミリーを使うための出発点として最適です。または音声機能や負荷分散機能を必要としない小規模運用環境に最適です。

サポートされるバージョン

この章の内容は、以下の製品バージョンに適用されます。

- AlarmPoint Java Client バージョン 3.2.1 以降
- AlarmPoint バージョン 3.2.1 以降
- NNMi バージョン 8.01 以降（NNMi 統合イネーブルメントライセンスが付属している場合）

NNMi と AlarmPoint は、異なるコンピュータにインストールすることをお勧めします。NNMi 管理サーバーと AlarmPoint サーバーのオペレーティングシステムは、同じでも、異なっていても構いません。

ドキュメント

HP NNMi–AlarmPoint 統合は、統合に付属している『*AlarmPoint 対応 HP Network Node Manager i-series 統合ガイド*』で、詳しく説明しています。

AlarmPoint のドキュメントスイートでは、AlarmPoint の機能について詳しく説明しています。ドキュメントスイートは、以下の AlarmPoint Customer Support サイトからダウンロードできます。

http://support.alarmpoint.com/

『*AlarmPoint Express for HP NNMi Quick Start Guide*』では、AlarmPoint の機能の概要を説明しています。このガイドでは、HP NNMi–AlarmPoint Express 統合のインストール方法、設定方法、管理方法を説明しています。

HP NNMi–AlarmPoint 統合を有効にする

AlarmPoint for NNMi 統合のインストール方法と設定方法の手順概要は、以下のとおりです。詳細は、*AlarmPoint 対応 HP Network Node Manager i-series 統合ガイド*を参照してください。

1. AlarmPoint Java Client を NNMi 管理サーバーにインストールします。
2. AlarmPoint Java Client 用の NNMi 固有の統合スクリプトをインストールします。
3. AlarmPoint Web サーバーと AlarmPoint Application サーバーに、Web Services Library をインストールします。
4. *オプション*: AlarmPoint Web サーバーに、NNMi 用の AlarmPoint サブスクリプションパネルをインストールします。
5. AlarmPoint Developer IDE を使って、NNMi 用の AlarmPoint アクションスクリプトをインストールします。
6. AlarmPoint Application サーバーに、統合音声ファイルをインストールします。
7. AlarmPoint に、イベントドメイン（および、オプションで、サブスクリプションドメイン）を設定します。
8. Web Services Client for NNMi を設定します。
9. AlarmPoint スクリプトをトリガーする、NNMi インシデントタイプを設定します。
10. 統合によって、AlarmPoint 通知に NNMi インシデントパラメータが挿入されることと、AlarmPoint が NNMi インシデントを正しく更新することを検証します。

HP NNMi–AlarmPoint 統合の使用法

NNMi と AlarmPoint は、相互動作して、ユーザーに通知を配信し、NNMi にその応答を戻します。NNMi は、ネットワーク内に問題点（たとえば、インシデント）を検出すると、以下の処理を行います。

1. NNMi は、問題点を説明するパラメータ（たとえば、影響を受けるコンピュータと状況）を伴って、AlarmPoint Client (APClient) を呼び出します。
2. APClient は、AlarmPoint エージェント (APAgent) に情報を送信します。
3. APAgent は問題点の詳細情報を AlarmPoint に送信し、AlarmPoint はその情報を適切な受信者に通知します。
4. 受信者はその通知に対して応答します。Web サービスコールを通じて、受信者による受諾、注釈付加、または優先度変更に従って、NNMi が更新されます。

HP NNMi–AlarmPoint 統合を無効にする

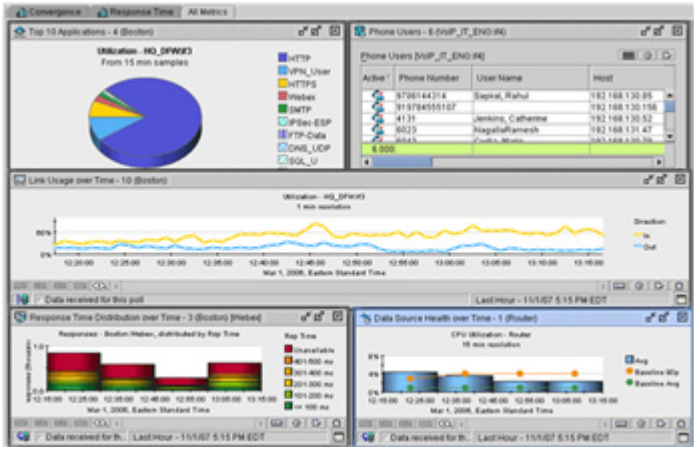
統合を無効にするには、統合の実行可能アーカイブによりインストールされたコンポーネントを削除します。

AlarmPoint の実装を削除する方法は、*AlarmPoint 対応 HP Network Node Manager i-series 統合ガイド*を参照してください。

HP NNMi–AlarmPoint 統合のトラブルシューティング

統合の最適化、拡張方法、および既知の問題は、『*AlarmPoint 対応 HP Network Node Manager i-series 統合ガイド*』を参照してください。

nGenius Performance Manager



NetScout Systems nGenius Performance Manager では、以下の目的のために、複雑なネットワークを見通すことができます。

- アプリケーションの認識とモニタリング
- パケットとフローの分析とトラブルシューティング
- 応答時間分析
- 報告および容量計画作成
- 集中性管理
- 警報生成とイベント識別

nGenius Performance Manager は、パケットの綿密な調査とフローベースの技術を活用し、いくつかのタイプのデータについて、リアルタイムの稼働インテリジェンスの見通しを提供します。

- 高度で重要なパフォーマンスインジケータ (KPI)。たとえば、応答時間、エラー、ジッターなど
- アプリケーションフローデータ。たとえば、使用率、会話、トップのトーカーなど
- パケットレベルの分析。たとえば、復号化、反発図など

nGenius Performance Manager では、広範囲のネットワークデータソースからパフォーマンスデータを収集し、すべてのネットワークインフラストラクチャ、トポロジ、およびアプリケーションの使用法パターンをモニタリングできます。次に、nGenius Performance Manager は、リアルタイムと履歴のビューおよび以下の情報を示すレポートの集合で結果を表示します。

- アプリケーションパフォーマンス
- ネットワークリソースのユーザーと悪用者
- ネットワーク容量のリソース消費

nGenius Performance Manager の購入については、HP 販売員にお尋ねください。

この章では、以下の内容を記載しています。

- HP NNMi–nGenius Performance Manager 統合
- HP NNMi–nGenius Performance Manager 統合の有効化
- HP NNMi–nGenius Performance Manager 統合の使用法
- HP NNMi–nGenius Performance Manager 統合の無効化

HP NNMi–nGenius Performance Manager 統合

nGenius Performance Manager を HP Network Node Manager i-series Software 環境に組み入れると、ネットワーク デバイスのモニタリングと管理のために NNMi を使用するネットワーク管理者は、nGenius デバイスによってアプリケーションレベルの見通しを得られます。

HP NNMi–nGenius Performance Manager 統合には、以下の機能が備わっています。

- nGenius Performance Manager サーバーを受信し、NNMi インシデントビューでアラームをプローブします。
- コンテキストビューを nGenius Performance Manager に起動して、インシデントの原因を調査します。
- NNMi マップビューの NetScout アイコンで nGenius プローブを表示します。
- NNMi コンソールから nGenius Performance Manager QuickViews を起動します。
- NNMi コンソールから nGenius Performance Manager アプリケーションを起動します。
- NNMi インシデントビューで利用できる nGenius Performance Manager インシデントについて NNMi レイヤー 2 およびレイヤー 3 の近隣接続ビューを起動します。
- NNMi インシデントビューで利用できる nGenius Performance Manager インシデント用に NNMi パスビューマップを起動します。
- nGenius Performance Manager が生成する各アラームについて NNMi にクリアトラップアラームを転送します。

値

HP NNMi–nGenius Performance Manager 統合には、以下の利点があります。

- 最大のネットワーク可用性を配信するコストを削減します。
- ネットワーク管理インフラストラクチャを 1 つのコンソールに統合します。
- 統合障害とアプリケーションの認識パフォーマンスデータによって、スタッフの生産性および効率性を向上させます。
- フローとパケットレベル詳細のコンテキストドリルダウンで MTTR を縮小し、パフォーマンス問題を識別します。

サポートされるバージョン

この章の情報は、以下の製品バージョンに当てはまります。

- nGenius Performance Manager バージョン 4.3 ビルド 450 以降
- nGenius K2 バージョン 1.0
- CDM Agent Firmware バージョン 4.3 ビルド 130 以降
- nGenius AFMon appliance バージョン 4.3 ビルド 555 以降

- nGenius InfiniStream Capture Engine バージョン 4.5 以降
- NNMi バージョン 8.02 以降

NNMi と nGenius Performance Manager サーバーは異なるコンピュータにインストールする必要があります。NNMi 管理サーバーと nGenius Performance Manager サーバーコンピュータは、同じオペレーティングシステムでも、異なるオペレーティングシステムでも構いません。

ドキュメント

HP NNMi–nGenius Performance Manager 統合については以下のドキュメントに詳細な説明があります。

nGenius Performance Manager v4.3 と HP Network Node Manager i ソフトウェアとの統合（NetScout パーツ番号 733-0089 Rev. A。次の URL から入手できます。**http://www.netscout.com**）

HP NNMi–nGenius Performance Manager 統合の有効化

HP Network Node Manager インストールファイル用の nGenius Performance Manager 統合ユーティリティは nGenius Performance Manager サーバーの次の場所で入手できます。

- *Windows*: `%nGenius Install%\rtm\bin\nGeniusNNM8.zip`
- *UNIX*: `$nGenius Install/rtm/bin/nGeniusNNM8.zip`

管理特権またはルート特権を持つユーザーが、NNMi 管理サーバーに統合ユーティリティをインストールします。ユーティリティは、nGenius サーバー用のすべての NNMi 統合関連設定データを NNMi にインポートします。

nGenius Performance Manager 統合ユーティリティをインストールする高度なステップは次のとおりです。詳細については、*nGenius Performance Manager v4.3 と HP Network Node Manager i ソフトウェアとの統合*を参照してください。

1. インストールファイルを抽出し、nGenius Performance Manager 内部で NNMi サポートを設定します。
2. NNMi で NetScout インシデント（アラーム）を設定します。
3. nGenius Probe が SNMP トラップをポート 395 に送信するように設定します。
4. *オプション。*プローブルーターマッピングを設定します。

HP NNMi–nGenius Performance Manager 統合の使用法

HP NNMi–nGenius Performance Manager 統合の使用法については、*nGenius Performance Manager v4.3 と HP Network Node Manager i ソフトウェアとの統合*を参照してください。

HP NNMi–nGenius Performance Manager 統合の無効化

HP NNMi–nGenius Performance Manager 統合の無効化については、*nGenius Performance Manager v4.3 と HP Network Node Manager i ソフトウェア*との統合を参照してください。

HP NNMi–nGenius Performance Manager 統合のトラブルシューティング

HP NNMi–nGenius Performance Manager 統合のトラブルシューティングについては、NetScout Systems 顧客サポートにお問い合わせください。問い合わせ情報は次の URL にあります。

http://www.netscout.com/support

NNM iSPI for Performance

この章では、NNM iSPI for Performance と NNMi を統合するための考慮事項とベストプラクティスについて説明します。

NNM iSPI for Performance の購入については、HP 販売員にお尋ねください。

この章では、以下の内容を記載しています。

- NNM iSPI for Performance とは何か
- NNM iSPI for Performance の有効化
- HA で実行されるよう NNM iSPI for Performance を設定
- NNM iSPI for Performance の使用法
- NNM iSPI for Performance のバックアップを規定
- NNM iSPI for Performance のリセット
- NNM iSPI for Performance を異なるコンピュータに移動
- NNM iSPI for Performance の無効化
- NNM iSPI for Performance のトラブルシューティング

NNM iSPI for Performance とは何か

NNM iSPI for Performance によって、データ収集、しきい値監視、HP Network Node Manager i Software 内部の MIB 変数の事前定義セットのしきい値警報生成ができます。NNMi から NNM iSPI for Performance レポートを起動できます。

NNMi パフォーマンスモニタリング設定によって、NNMi がパフォーマンスデータを収集するインタフェース、およびデータ収集間隔が決定されます。NNM iSPI for Performance は NNMi データを受信および分析し、分析結果を独占的なフォーマットで保存します。

NNM iSPI for Performance は NNMi とは別にライセンス契約されます。しかし、両方の製品とも同じノードカウントについてライセンス契約される必要があります。

値

NNM iSPI for Performance は、順位付け、傾向判定、および例外報告について NNMi にパフォーマンス報告を追加します。

- 順位付けは、使用率ごとに要素をソートし、使用率が異常に高いか、または低いネットワーク要素を強調表示します。
- 傾向判定は、期間を調べ、パフォーマンスが安定しているか、悪化しているか、または改善しているかをカラーコード化で示します。
- 例外報告は、すべてのしきい値状況を調べ、時間の経過とともにしきい値状況を追跡します。

サポートされるバージョン

この章の情報は、以下の製品バージョンに当てはまります。

- NNM iSPI for Performance バージョン 8.10 以降
- NNMi バージョン 8.10 以降

ドキュメント

NNM iSPI for Performance リリースノート、対応マトリックス、およびインストールの指示は、次の URL から入手できます。

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

さらに、NNM iSPI for Performance ヘルプが製品とともにインストールされています。

NNM iSPI for Performance の有効化

NNM iSPI for Performance は、NNMi 管理サーバーまたは専用サーバーにアドオン製品としてインストールできます。推奨するインストールモデルは、パフォーマンスポーリングの頻度とボリュームで決まります。詳細は、NNM iSPI for Performance 対応マトリックスを参照してください。

インストールを計画するときは、ディスク容量要件を追加し、単一コンピュータにインストールする予定の全ソフトウェアの RAM 要件を確認することを忘れないでください。

以下のステップに、NNM iSPI for Performance を有効化するための高度なプロセスを示します。

1. 各 NNMi 管理サーバーで、NNM iSPI for Performance 有効化スクリプトを実行します。

   詳細は、『*NNM iSPI for Performance インストールガイド*』を参照してください。
2. 『*NNM iSPI for Performance インストールガイド*』の説明に従って、NNM iSPI for Performance をインストールします。
3. NNMi コンソールで、パフォーマンスモニタリングを設定します。

   詳細は、NNM iSPI for Performance ヘルプの「*Setting Polling Characteristics in NNMi（NNMi でポーリングの特徴を設定）*」と「*Setting Thresholds for Metrics（メトリック用しきい値の設定）*」を参照してください。

4 NNM iSPI for Performance で、パフォーマンス報告を設定します。

詳細は、NNM iSPI for Performance ヘルプの「*スケジューリングレポート*」を参照してください。

HA で実行されるよう NNM iSPI for Performance を設定

高可用性 (HA) で NNMi を実行する計画の場合は、次のオプションが利用できます。

- NNM iSPI for Performance を NNMi 管理サーバーにインストールする場合は、iSPI を NNMi と同じ HA リソースグループ内で実行する必要があります。この設定の詳細は、NNM iSPI for Performance をアドオン iSPI として設定ページの 89 を参照してください。
- NNM iSPI for Performance が専用サーバーにある場合、iSPI は、NNMi HA クラスタ内で別々の HA リソースを実行できますし、または HA 外で実行することもできます。いずれの場合も、iSPI は NNMi HA リリースグループの仮想ホスト名に接続します。HA シナリオの詳細は、専用サーバーで HA 用に NNM iSPI for Performance を設定するページの 91 を参照してください。

NNMi HA 構成の一部として、NNM iSPI for Performance 有効化スクリプトをクラスタ内の各 NNMi 管理サーバーで実行する必要があります。

NNM iSPI for Performance は、`PerfSPIHA` Perl モジュールを配信します。このモジュールには、HA 設定コマンドのカスタム実装が含まれています。HA 設定コマンドを実行するときは、命令の中で *<iSPI_PM_Name>* の代わりに値 `PerfSPIHA` を使用します。

NNM iSPI for Performance の使用法

NNM iSPI for Performance は、NNMi コンソールにいくつかの報告アクションを追加しています。これらの各アクションは、指定のレベルで NNM iSPI for Performance ユーザーインタフェースを起動します。

NNM iSPI for Performance レポートの詳細は、NNM iSPI for Performance ヘルプを参照してください。

NNM iSPI for Performance のバックアップを規定

NNMi バックアップスクリプトは、パフォーマンスモニタリング設定をバックアップするのみです。NNMi には、パフォーマンスデータをバックアップするメカニズムは備わっていません。

パフォーマンスデータをバックアップするには、NNM iSPI for Performance システムで別々のバックアップツールを選択および実装する必要があります。導入フォルダとそのすべてのサブフォルダを別々のバックアッププロセスに追加します。

NNM iSPI for Performance は次のフォルダにデータを保存します。

- *Windows*: `%NnmInstallDir%\NNMPerformanceSPI\deploy\`
- *UNIX*: `$NnmInstallDir/NNMPerformanceSPI/deploy/`

NNM iSPI for Performance のリセット

空の状態から再開始するできるように NNMi 設定とデータベースを完全にリセットする場合は、NNM iSPI for Performance 設定もリセットして、2 つの製品の間の協定でトポロジ識別子を保持する必要があります。

NNMi データベースをリセットした後で、NNM iSPI for Performance ヘルプの「*Resetting the Performance SPI（パフォーマンス SPI のリセット）*」に説明してあるように NNM iSPI for Performance リセットユーティリティを実行します。

NNM iSPI for Performance を異なるコンピュータに移動

NNM iSPI for Performance を NNMi テスト環境に含めることができます。`nnmconfigexport.ovpl` コマンドは、生産 NNMi 管理サーバーでテスト設定を複製できるように、NNMi パフォーマンスモニタリング設定をエクスポートします。

NNM iSPI for Performance レポートは、NNMi トポロジ内のオブジェクトの一意の識別子を基礎にしています。これらの識別子は新しい NNMi 管理サーバーでは異なってくるので、テストシステムから製造システムに既存のレポートを移植することはできません。

テストシステムから製造環境に NNMi を移動するシナリオの詳細は、NNMi 管理サーバーの変更ページの 131 を参照してください。

NNM iSPI for Performance の無効化

NNM iSPI for Performance のアンインストールの詳細は、NNM iSPI for Performance ヘルプの「*Uninstalling the Performance SPI（パフォーマンス SPI のアンインストール）*」を参照してください。

NNM iSPI for Performance のトラブルシューティング

NNM iSPI for Performance のアンインストールについては、NNM iSPI for Performance ヘルプの「*Troubleshooting（トラブルシューティング）*」を参照してください。

追加情報

この項では以下の付録について説明します。

- NNMi 環境変数
- Causal Engine と NNMi インシデント

NNMi 環境変数

HP Network Node Manager i Software には、ファイルシステム内の移動やスクリプトの作成に使用できる多数の環境変数があります。

この付録では、以下の内容を記載しています。

- このドキュメントで使用する環境変数
- 他の使用可能な環境変数
- NNMi 8.00 からの Windows パスおよび環境変数

このドキュメントで使用する環境変数

このドキュメントでは、以下の 2 つの NNMi 環境変数を使用して、ファイルやディレクトリの場所を参照します。デフォルト値は次のとおりです。実際の値は、NNMi のインストール中に行った選択内容によって異なります。

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

➤ Windows システムでは、NNMi インストールプロセスによってこれらの環境変数が作成されるので、いつでも使用できます。

➤ 最初に NNMi 8.00 をインストールした場合、ご使用のシステムでは、NNMi 8.00 からの Windows パスおよび環境変数ページ 238 で説明しているようにこれらの環境変数で異なる値を使用しています。

- UNIX:
  - `$NnmInstallDir`: `/opt/OV`
  - `$NnmDataDir`: `/var/opt/OV`

➤ UNIX システムでは、これらの環境変数を使用する場合は手動で作成する必要があります。

他の使用可能な環境変数

NNMi 管理者は、いくつかの NNMi ファイルの場所には定期的にアクセスします。NNMi には、頻繁にアクセスする場所へ移動するための多数の環境変数を設定するスクリプトがあります。

最初に NNMi 8.00 をインストールした場合は、NNMi 8.00 からの Windows パスおよび環境変数ページ 238 を参照してください。

NNMi 環境変数の拡張リストをセットアップするには、次の例のようなコマンドを使用します。

- Windows: "`C:\Program Files\HP\HP BTO Software\bin\nnm.envvars.bat`"
- UNIX: `/opt/OV/bin/nnm.envvars.sh`

上記の各 OS 用のコマンドを実行した後で、表 22 (Windows) または 表 23 (UNIX) で示す NNMi 環境変数を使用して、頻繁に使用する NNMi ファイルの場所に移動できます。

表 22　Windows OS での環境変数の場所

変数	Windows（例）
%NNM_BIN%	C:\Program Files\HP\HP BTO Software\bin
%NNM_CONF%	C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\conf
%NNM_DATA%	C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software
%NNM_DB%	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
%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\
%NNM_LOG%	C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\log
%NNM_PRIV_LOG%	C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\log

表 22　**Windows OS** での環境変数の場所

変数	Windows（例）
%NNM_SHARE_LOG%	C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\log
%NNM_SNMP_MIBS%	C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\share\snmp_mibs
%NNM_SUPPORT%	C:\Program Files\HP\HP BTO Software\support
%NNM_TMP%	C:\Documents and Settings\All Users\Application Data\HP\HP BTO Software\tmp
%NNM_WWW%	C:\Program Files\HP\HP BTO Software\www

表 23　**UNIX OS** での環境変数の場所

変数	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_PRIV_LOG	/var/opt/OV/log
$NNM_SHARE_LOG	/var/opt/OV/log
$NNM_SNMP_MIBS	/var/opt/OV/share/snmp_mibs
$NNM_SUPPORT	/opt/OV/support
$NNM_TMP	/var/opt/OV/tmp
$NNM_WWW	/opt/OV/www

NNMi 8.00 からの Windows パスおよび環境変数

Windows OS では、NNMi ファイルへのデフォルトパスは、バージョン 8.00 と 8.01（以上）では異なります。NNMi 8.00 より上の新バージョンの NNMi にアップグレードすると、新しいバージョンの NNMi は NNMi 8.00 で定義されたパスを使い続けます。この場合、NNMi インストール中に作成された環境変数の場所は以下のようになります。

- `%NnmInstallDir%: <drive>\Program Files (x86)\HP OpenView`
- `%NnmDataDir%: <drive>\Program Files (x86)\HP OpenView\data`

NNMi 環境変数の拡張リストをセットアップするには、次の例のようなコマンドを使用します。

```
"C:\Program Files(x86)\HP OpenView\bin\nnm.envvars.bat"
```

コマンドを実行した後で、表 24 で示す NNMi 環境変数を使用して頻繁に使用する NNMi ファイルの場所へ移動できます。

➤ アップグレードのみの場合は、表 24 の情報を 236 ページの表 22 の情報より優先してください。

表 24　NNMi 8.00 環境変数の場所（Windows OS の場合）

変数	Windows（例）
%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
%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_PRIV_LOG%	C:\Program Files (x86)\HP OpenView\data\log
%NNM_SHARE_LOG%	C:\Program Files (x86)\HP OpenView\data\log

表 24　NNMi 8.00 環境変数の場所（Windows OS の場合）

変数	Windows（例）
%NNM_SNMP_MIBS%	C:\Program Files (x86)\HP OpenView\data\share\snmp_mibs
%NNM_SUPPORT%	C:\Program Files (x86)\HP OpenView\support
%NNM_TMP%	C:\Program Files (x86)\HP OpenView\data\tmp
%NNM_WWW%	C:\Program Files (x86)\HP OpenView\www

Causal Engine と NNMi インシデント

通信とデータネットワークは規模と複雑さが著しく伸び、発生する障害の数も増えています。障害が 1 つ発生しただけでもたくさんの警報が発生することもあり、ネットワークオペレータにとっての障壁は、あらゆる逸話的警報の中から本当の問題を見分けることになりました。従来のイベント相関システムにより警報の数を減らすことは可能でしたが、これらのシステムは根本原因を自動化された方法で突き止めるという点で劣る傾向があります。

HP Network Node Manager i Software Causal Engine 技術は、因果関係ベースのアプローチを使用して、**根本原因解析 (RCA)** をネットワーク症状に適用します。

この付録では、以下の内容を記載しています。

- 因果関係解析 - 高度な考察
- Causal Engine の概念
- ステータスの概念
- エピソードとは
- NNMi は何を解析するのか？
- 失敗のシナリオは何ですか？
- ネットワーク設定の変更
- NNMi 管理設定の変更

因果関係解析 - 高度な考察

Causal Engine 技術により、以下の高度な機能が可能になります。

- NmsApa jboss サービスを使用して、ネットワークを解析する
- RCA へのモデルベースのアプローチ
  - 管理対象オブジェクト同士の間の行動的関連をモデル化する
  - イベント因果関係に加えてオブジェクトモデルを使用して解析を進める
  - 根本原因と影響を判定する
  - MINCAUSE アルゴリズムをベースにする
  - あいまい性および部分的症状に対処可能である
- 動的
  - 解析中に症状を積極的に誘発させる
  - トポロジの変化に動的に反応する
- 拡張性
  - モジュールの階層を採用（インポート / エクスポート）
  - ネットワーク障害のエンドツーエンドの診断を提供する
  - 将来の製品でのルールセット追加を可能にする

Causal Engine の概念

Causal Engine 技術では、以下の逐次的アプローチを使用します。

1 根本原因問題と症状を形式的に定義します。

2 モデルを使用して症状を根本原因問題に関連付けることにより、解析を行います。症状の源は、以下の 2 つです。

- StatePoller（症状が状態の変化の場合）。
- イベント（症状がトラップの場合）。

3 根本原因に関連する結論を生み出します。

Causal Engine の結論には、モデルに関連したアーチファクト（成果物）が含まれています。アーチファクトには、以下の詳細が含まれます。

- インシデント発生
- インシデント相関
- インシデント抑制
- インシデント中止
- 関連するオブジェクトのステータス

ステータスの概念

インシデント操作に加えて、NmsApa サービスは関連オブジェクトのステータスを設定します。ステータスはオブジェクトの状態全般を示すために使用され、未解決結論の結果として計算されます。どの結論にも重大度が関連付けられてあり、報告されるステータスはすべての未解決結論のうちで最も厳格です。さらに、結論はユーザーに、オブジェクトのステータスについての根本*原因*（つまり理由）を知らせます。

NmsApa サービスは、以下のオブジェクトを管理します。

- SNMP エージェント
- IPv4 アドレス
- インタフェース
- 接続
- ノード
- ノードグループ

NmsApa サービスは、重大度の高いものから順に以下のステータスカテゴリを使用します。

- 不明
- 使用不可
- 危険域
- 警戒域
- 注意域

- 正常域
- ステータスなし

エピソードとは

NmsApa サービスの目標は、オペレータやネットワークエンジニアが対処できるたった 1 つのインシデントを提示することです。そのために、NmsApa サービスは**エピソード**の概念を使用します。エピソードは特定の期間存在し、その間に 2 番目の障害は設定に基づいて相関付けられるか抑制されます。例：

- AddressNotResponding インシデントは、InterfaceDown インシデントにより、以下のシナリオに従って抑制されます。
  - IPv4 アドレスが ICMP への応答を停止すると、エピソードが開始して 60 秒間存続します。
  - その期間内に、IPv4 アドレスに関連付けられたインタフェースが停止すると、NmsApa サービスは*インタフェース停止*状態が原因で IPv4 アドレスが応答を停止したと結論付けます。
  - したがって、AddressNotResponding インシデントは発生しません。InterfaceDown インシデントのみが発生します。
  - InterfaceDown インシデントがその期間内に検出されるようにするために、NmsApa サービスは*指定ポーリング*をそのインタフェースに対して発行します。これによりネットワークエンジニアは、問題の根本原因（この場合はインタフェースです）を修正可能になります。
  - インタフェースがエピソード中に停止しない場合、NmsApa サービスは AddressNotResponding インシデントを発生します。インタフェースがエピソード後に停止すると、InterfaceDown インシデントが発生します。この場合、ネットワークエンジニアは 2 つの問題に個々に対処しなければなりません。
- NodeDown インシデントは、1 ホップネイバー（隣接）インタフェースからの InterfaceDown インシデントを、次のシナリオに従って相関付けします。
  - インタフェースが停止すると、NodeDown エピソードが隣接ノードに対して開始され、900 秒間存続します。
  - その期間内に、ノードが停止すると、InterfaceDown インシデントが NodeDown インシデントの下で相関付けされます。
  - すべての 1 ホップネイバーからの InterfaceDown インシデントが NodeDown インシデントの下に相関付けされます。InterfaceDown インシデントを、NodeDown インシデントを裏付ける証拠として検討することができます。

NNMi は何を解析するのか？

NNMi は SNMP プロトコルを使用して管理対象ノードから情報を、SNMP エージェント（管理対象ノードで稼働しているプロセスで、管理機能を提供する）を使用して取得します。SNMP エージェントは、管理対象ノード上のインタフェースおよびポートを管理し、1 つ以上のノードと関連付けが可能です。

SNMP エージェントに関連付けられた可能な NNMi ステータスカテゴリの一覧を以下に示します。

- 不明 - 適用不可。
- 使用不可 - 適用不可。
- 危険域 - SNMP エージェントは SNMP クエリーに応答しません。
- 警戒域 - 適用不可。
- 注意域 - 適用不可。
- 正常域 - SNMP エージェントは SNMP クエリーに応答します。
- ステータスなし - SNMP エージェントはポーリングされません。

IPv4 アドレスは、ICM に応答するルーティング可能なアドレスです。IPv4 アドレスは、通常はノードに関連付けされます。NNMi は、ノードのステータスを次のように報告します。

- 不明 - 適用不可。
- 使用不可 - この IPv4 アドレスに関連付けられたインタフェースは管理不可能または使用不可にされています。
- 危険域 - IPv4 アドレスは ICMP クエリーに応答しません (デバイスを ping します)。
- 警戒域 - 適用不可。
- 注意域 - 適用不可。
- 正常域 - IPv4 アドレスは ICMP クエリーに応答します。
- ステータスなし - IPv4 アドレスはポーリングされません。

インタフェースとは、ノードをネットワークに接続するために使用する物理的なポートです。NNMi はインタフェースのステータスを次のように報告します。

- 不明 - インタフェースに関連付けられた SNMP エージェントは、SNMP クエリーに応答しません。不明は、`NmsApa` サービスが、`ifAdminStatus` と `ifOperStatus` を測定できないため、稼働状況を判定できないことを示します。
- 使用不可 - インタフェースは管理不可能です (`ifAdminStatus` = down)。
- 使用不可 - インタフェースは操作できません (`ifOperStatus` = down)。
- 警戒域 - 適用不可。
- 注意域 - 適用不可。
- 正常域 - インタフェースは操作可能です (`ifOperStatus` = up)。
- ステータスなし - インタフェースはポーリングされていません。

ノードとは、NNMi がスパイラル検出プロセスの結果として見つけ出すデバイスです。ノードには、インタフェース、ボード、およびポートを含むことができます。ノードは、次の 2 つのカテゴリに分けることができます。

1. ネットワークノード : スイッチ、ルーター、ブリッジおよびハブなどのアクティブデバイス
2. エンドノード : UNIX サーバーや Windows サーバーなど

NNMi は通常はネットワークノードを管理し、ノードステータスを次のように報告します。

- 不明 - ノードに関連した SNMP エージェントは SNMP クエリーに応答せず、ポーリングした IPv4 アドレス は ICMP クエリーに応答しません。これは、NNMi がノードを管理できないことを示します。
- 使用不可 - 適用不可。

- 危険域 - 次のいずれかになります。
  - ノードは、隣接解析の決定により停止しています。
  - ノードは重要とマークされており、*管理が困難です*（NNMi はノードに NNMi サーバーからアクセスできません）。
  - ノードはアイランドであり（ネイバーがない）、そのため管理が困難です。
  - `NmsApa` サービスは、ノードが停止しているか、または着信接続が停止しているかを判定できません。
- 警戒域 - 次のいずれかになります。
  - ノードに関連付けられた SNMP エージェントは、SNMP クエリーに応答しません。
  - ノード内の 1 つ以上のインタフェースが停止しています。
  - ノード上の 1 つ以上の IPv4 アドレスが ICMP に応答していません。
- 注意域 - 適用不可。
- 正常域 - ノードの SNMP エージェント、ポーリングしたインタフェース、およびポーリングした IPv4 アドレスは稼働しています。
- ステータスなし - ノードの SNMP エージェント、すべてのインタフェース、およびすべての IPv4 アドレスはポーリングされていません。

接続はレイヤ 2 物理接続とレイヤ 3 ネットワーク接続です。NNMi は、転送データベース (FDB) 表を他のネットワークデバイスから読み取り、CDP や EDP などの検出プロトコルをサポートするデバイスを使用することにより、接続情報を検出します。NNMi は接続のステータスを、次のように報告します。

- 不明 - 接続のすべてのエンドポイントが不明なステータスを持っています。
- 使用不可 - 接続のいずれか 1 つのエンドポイントが使用不可です。
- 危険域 - すべてのエンドポイントは操作できません。
- 警戒域 - エンドポイントのいずれか 1 つが停止しています。
- 注意域 - エンドポイントは、不明だが危険でないステータスを持っています。
- 正常域 - すべてのエンドポイントは、操作可能です。
- ステータスなし - いずれか 1 つのエンドポイントがポーリングされません。

ノードグループはノードの論理的コレクションで、ポーリング設定を分離するために使用します。管理者は、ノードタイプのグループ化を作成します。たとえばルーターなど一部のノードは業務上絶対不可欠であるため、これらのルーターはより頻繁にポーリングするのがよいでしょう。そのためには、重要なルーターが入ったノードグループを定義して、これらのグループにより短いポーリングサイクルを設定します。

NNMi は、ノードグループのステータスを次のように報告します。

- 不明 - グループ内のすべてのノードが不明なステータスを持っています。
- 使用不可 - 適用不可。
- 危険域 - グループ内のすべてのノードが危険なステータスを持っています。
- 警戒域 - グループ内の 1 つ以上のノードが危険なステータスを持っています。
- 注意域 - ノードは、不明だが危険でないステータスを持っています。
- 正常域 - グループ内のすべてのノードが正常なステータスを持っています。
- ステータスなし - グループ内のすべてのノードがステータスを持っていません。

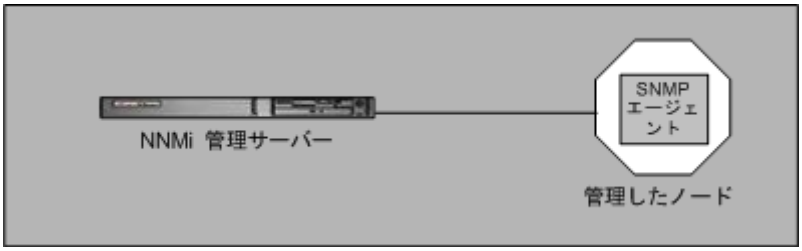
失敗のシナリオは何ですか？

以下のシナリオは、ネットワークにおける問題の例と、Causal Engine がこれらの問題を診断するために行う作業を示しています。これらのシナリオが示すインシデントの例を他の例と共に 表 25 で示します。

表 25　インシデントの定義

インシデント名	説明
AddressNotResponding	IPv4 アドレスは ICMP に応答していません。次の理由が考えられます。 1 ノードが停止している。 2 デバイス（ルーターなど）の設定に誤りがあるため、いくつかの IPv4 アドレスに到達できない。
InterfaceDown	インタフェースがポーリングに応答していません。
ConnectionDown	接続の末端部の両方（またはすべて）が停止しています。
ConnectionPartiallyUnresponsive	このインシデントは、接続が部分的に無反応であることを示しています。
NodeDown	このインシデントは、`NmsApa` サービスが次の解析に基づいてノードが停止していると判定したことを示しています。 • このノードに割り当てられている IPv4 アドレスの 100% が到達不可能である。 • このマシンにインストールされている SNMP エージェントが応答していない。少なくとも 2 つの隣接デバイスが到達可能であり、このノードへの接続性について問題を報告している。
NodeOrConnectionDown	このインシデントは、ノードが ICMP または SNMP クエリーに応答していないことを示します。また、隣接ノードが 1 つだけ停止しているため、ノードが停止しているのか接続が停止しているのか `NmsApa` サービスが判断できないことを示しています。

SNMP エージェントが SNMP クエリーに応答しない



説明

管理対象ノード	ネットワークデバイス (Ethernet スイッチなど)
SNMP エージェント	管理対象ノード用の新しいコミュニティ文字列あり
MS 通信設定	新しいコミュニティ文字列での更新なし

シナリオ : SNMP エージェントが応答していません。たとえば、この SNMP エージェントのコミュニティ文字列が変更され、NNMi の通信設定がまだ更新されていないけれども、ノードが稼働しています（IPv4 アドレスを ping 可能です)。

根本原因：SNMP エージェントが応答していません。

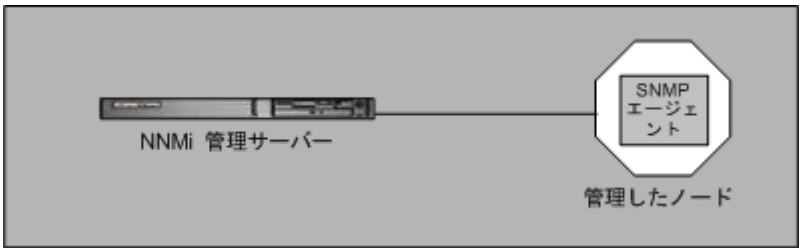
インシデント：発生なし。

ステータス：SNMP エージェントが危険な状態です。

結論：`SNMPAgentNotResponding`

結果：ノード ステータスは警戒域であり、ノードについての結論は `UnresponsiveAgentInNode` です。ポーリングされたすべてのインタフェースは、NNMi により管理できないため、不明ステータスです。各インタフェースについての結論は `InterfaceUnmanageable` です。

SNMP エージェントが SNMP クエリーに応答している



説明

管理対象ノード	ネットワークデバイス (Ethernet スイッチなど)
SNMP エージェント	管理対象ノード用の新しいコミュニティ文字列あり
MS 通信設定	新しいコミュニティ文字列で更新済

シナリオ：ページ 247 を参照してください。このシナリオは、*SNMP エージェントが SNMP クエリーに応答しない*シナリオに続いています。NNMi 管理者が通信設定を更新して新しいコミュニティ文字列を含めることを想定します。管理対象ノードの SNMP エージェントが SNMP クエリーへの応答を開始します。

根本原因：SNMP エージェントが応答していません。

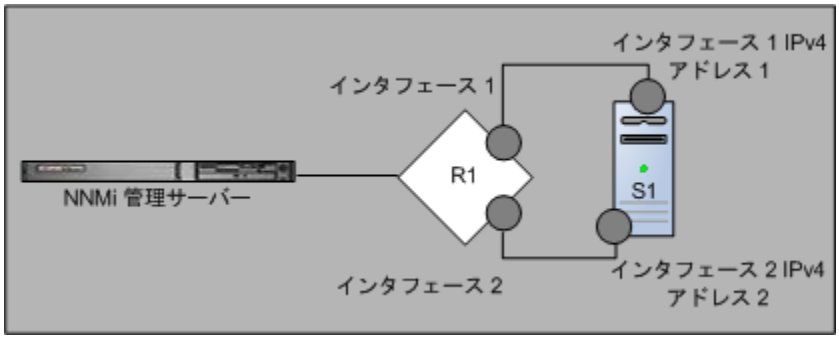
インシデント：発生なし。

ステータス：SNMP エージェントは正常な状態です。

結論：`SNMPAgentResponding`

結果：ノードステータスは正常域であり、ノードについての結論は `ResponsiveAgentInNode` です。`InterfaceUnmanagable` はポーリングされたすべてのインタフェースから除去されて、インタフェースは前のステータスに戻ります。

IPv4 アドレスが ICMP に応答しない



説明

R1	ルーター 1
経路	ルーター上でインタフェース 1 からインタフェース 2 に変更された
S1	サーバー 1
管理対象ノード S1	マルチホームサーバー
S1 インタフェース 1	IPv4 アドレス 1 と関連
S1 インタフェース 2	IPv4 アドレス 2 と関連

シナリオ：S1 の IPv4 アドレス 1 が応答していません。たとえば、ルーター 1 (R1) の経路がインタフェース 1 からインタフェース 2 に変わったことにより、S1 のインタフェース 1 を宛先としていたパケットが現在は R1 のインタフェース 2 からルーティングされていると想定します。関連付けられているインタフェースは稼働しており、いくつかの IPv4 アドレスを ping できるので、ノードは到達可能です。SNMP エージェントは稼働しています。

根本原因：IPv4 アドレスが応答していません。

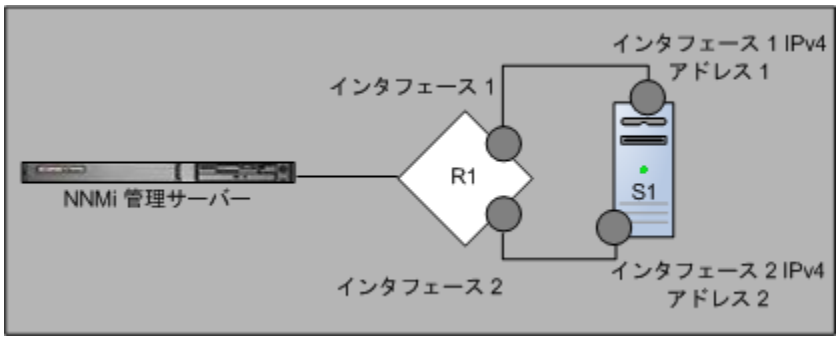
インシデント：`AddressNotResponding` インシデントが発生しました。

ステータス：IPv4 アドレス は危険な状態です。

結論：`AddressNotResponding`

結果：ノードステータスは警戒域であり、ノードについての結論は `SomeUnresponsiveAddressesInNode` です。

ICMP への IPv4Address 応答



説明

R1	ルーター 1
経路	ルーター上でインタフェース 2 からインタフェース 1 に変更された
S1	サーバー 1
管理対象ノード S1	マルチホームサーバー
S1 インタフェース 1	IPv4 アドレス 1 と関連
S1 インタフェース 2	IPv4 アドレス 2 と関連

シナリオ：ページ 249 を参照してください。このシナリオは、*IPv4Address が ICMP に応答しない*シナリオに続いています。IPv4 アドレスが現在は応答しており、関連付けられたインタフェースが稼働しており、ノードに到達可能であることを想定してください（たとえば、いくつかの IPv4 アドレスを ping できたり、SNMP エージェントが稼働しているとします）。

根本原因：IPv4 アドレスが応答しています。

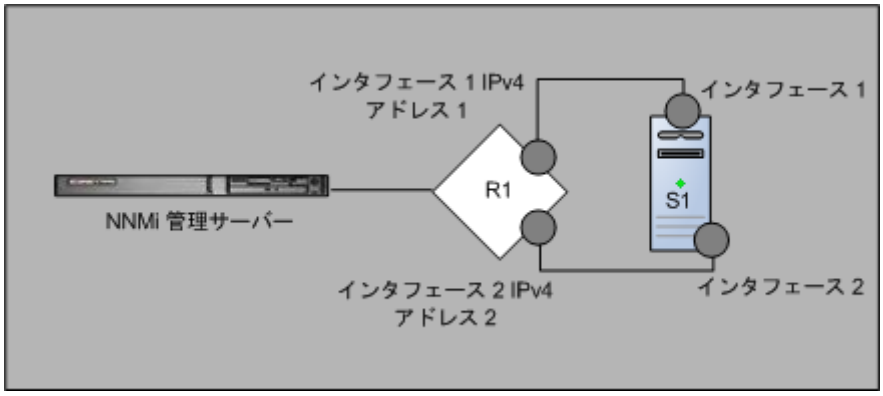
インシデント：発生なし。`AddressNotResponding` インシデントがクローズしました。

ステータス：IPv4 アドレスは正常な状態です。

結論：`AddressResponding`

結果：ノードステータスは正常域であり、ノードについての結論は `ResponsiveAddressesInNode` です。

インタフェースを操作できない



説明

R1	ルーター 1
R1 のインタフェース 1	管理可能であるが操作できないように設定されています
R1 のインタフェース 1	IPv4 アドレス 1 に設定されています
ルーター 1 の インタフェース 2	IPv4 アドレス 2 に設定されています
S1	サーバー 1

シナリオ：R1 インタフェース 1 は操作できず (`ifOperStatus` = down)、管理可能 (`ifAdminStatus` = up) です。R1 は `linkDown` トラップを送信します。R1 は到達可能です。いくつかの IPv4 アドレス（IPv4 アドレス 2 など）を ping できるためです。SNMP エージェントは稼働しています。IPv4 アドレス 1 はインタフェース 1 に関連付けられており、ICMP への応答を停止しました。

根本原因：インタフェースは停止しています。

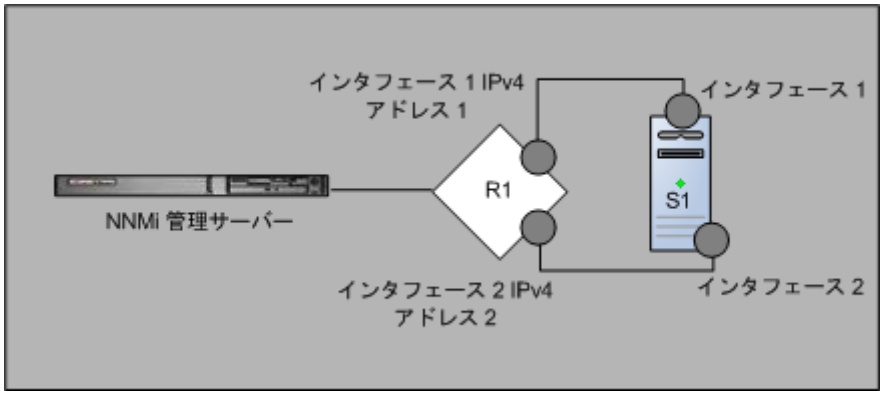
インシデント：`InterfaceDown` インシデントが発生しました。`LinkDown` インシデントが `InterfaceDown` インシデントの下に相関付けされています。

ステータス：インタフェースは危険な状態です。

結論：`InterfaceDown`

結果：ノードステータスは警戒域であり、ノードについての結論は `InterfacesDownInNode` です。`AddressNotResponding` インシデントが IPv4 アドレスに関連付けられていません。

インタフェースは操作可能である



説明

R1	ルーター 1
R1 のインタフェース 1	管理可能であり操作可能であるように設定されています
R1 のインタフェース 1	IPv4 アドレス 1 に設定されています
ルーター 1 の インタフェース 2	IPv4 アドレス 2 に設定されています
S1	サーバー 1

シナリオ：ページ 251 を参照してください。このシナリオは、*インタフェースを操作できないシナリオ*に続いています。R1 インタフェース 1 が現在は操作可能であると想定します (`ifOperStatus` = up)。ノードは到達可能です。その IPv4 アドレスをすべて ping できます。SNMP エージェントは稼働しています。

根本原因：インタフェースは稼働しています。

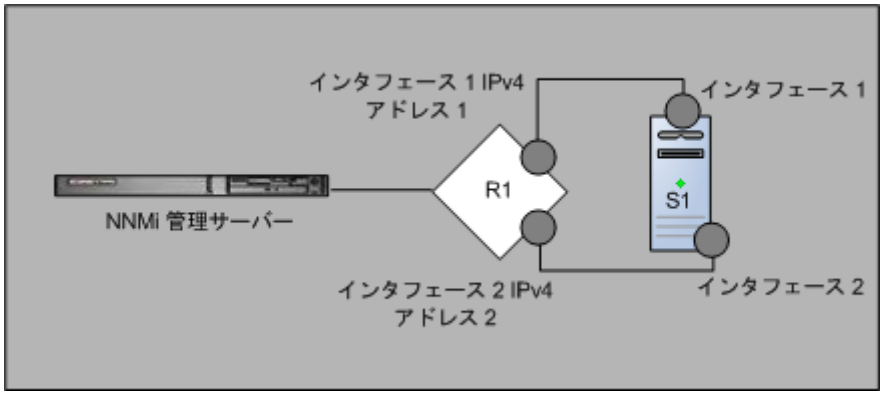
インシデント：発生なし。`InterfaceDown` インシデントがクローズしました。

ステータス：インタフェースは正常な状態です。

結論：`InterfaceUp`

結果：ノードステータスは正常域であり、ノードについての結論は `InterfacesUpInNode` です。

インタフェースを管理できない



説明

R1	ルーター 1
R1 のインタフェース 1	管理可能であり操作できないように設定されています
R1 のインタフェース 1	IPv4 アドレス 1 に設定されています

シナリオ : R1 インタフェース 1 は管理できません (`ifAdminStatus` = down) が、ノードは到達可能です。たとえば、インタフェース 2 を ping して SNMP エージェントが稼働していると想定します。R1 インタフェース 1 を無効にすると、そのインタフェースが操作できなくなります。このインタフェース IPv4 アドレス 1 に関連付けられた IPv4 アドレスが ICMP への応答を停止します。

根本原因 : R1 インタフェース 1 は使用不可です。

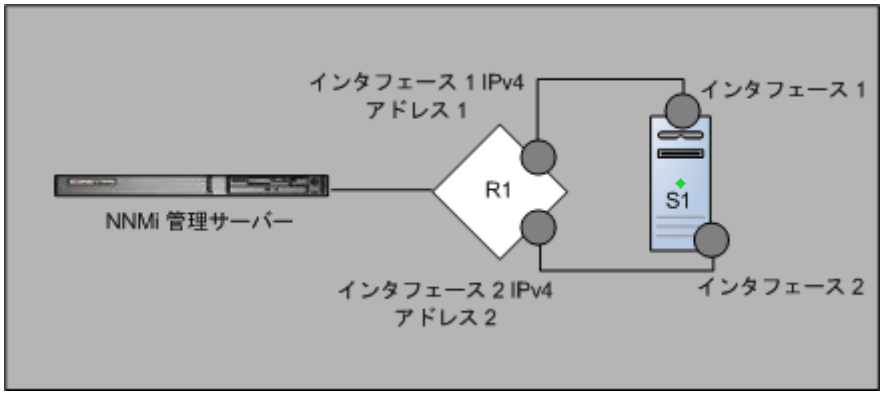
インシデント : 発生なし。

ステータス : インタフェースは使用不可の状態です。

結論 : `InterfaceDisabled`

結果 : R1 インタフェース 1 に関連付けられた IPv4 アドレスはステータスが使用不可です。IPv4 アドレスについての結論は `AddressDisabled` です。

インタフェースを管理できる



説明

MS	NNMi 管理サーバー
R1	ルーター 1
R1 のインタフェース 1	管理可能であり操作可能であるように設定されています
R1 のインタフェース 1	IPv4 アドレス 1 に設定されています

シナリオ：ページ 253 を参照してください。このシナリオは、*インタフェースを操作できないシナリオ*に続いています。R1 インタフェース 1 が現在管理可能であり (`ifAdminStatus` = up)、そのインタフェースのいくつかの IPv4 アドレスを ping することによりこのノードに到達できると想定します。SNMP エージェントは稼働しています。R1 インタフェース 1 を有効にすることにより、操作可能になります。このインタフェースに関連付けられた IPv4 アドレスが ICMP への応答を開始します。

根本原因：インタフェースは有効です。

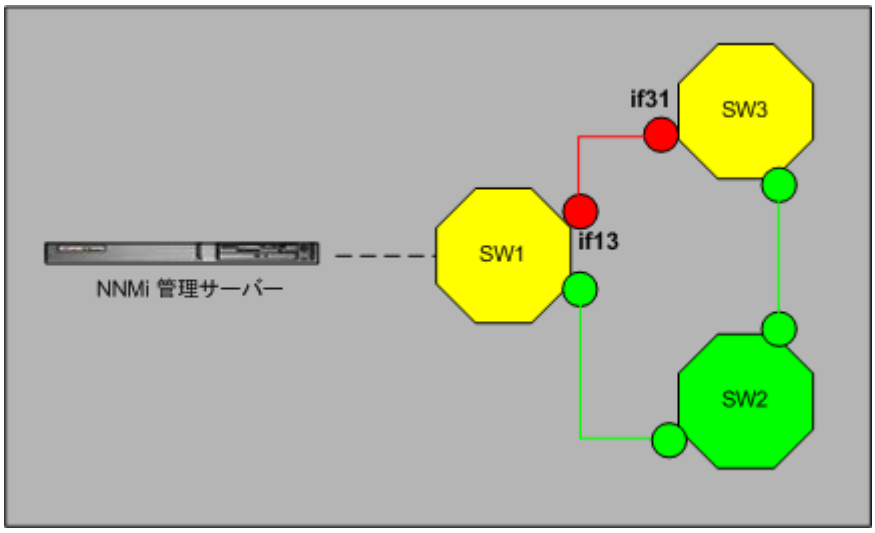
インシデント：発生なし。

ステータス：インタフェースは正常な状態です。

結論：`InterfaceEnabled`

結果：R1 インタフェース 1 に関連付けられた IPv4 アドレスはステータスが有効です。IPv4 アドレスについての結論は `AddressEnabled` です。

接続を操作できない



説明

SW1	スイッチ 1
SW2	スイッチ 2
SW3	スイッチ 3
IF31	スイッチ 1 に接続しているスイッチ 3 のインタフェース
IF13	スイッチ 3 に接続しているスイッチ 1 のインタフェース

シナリオ：スイッチ 1 (IF13) に接続しているスイッチ 3 のインタフェースと、スイッチ 3 (IF31) に接続しているスイッチ 1 のインタフェースとの間の接続が停止しています。トラフィックは、管理サーバーからスイッチ 1 (SW1) とスイッチ 2 (SW2) を通って流れます。IF13 と IF31 の両方が停止とマークされます。

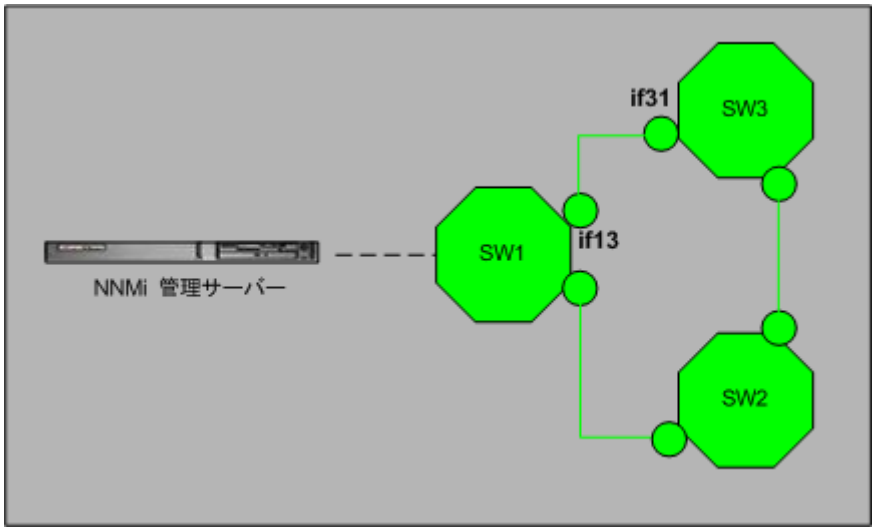
根本原因：IF13 と IF31 の間の接続が停止しています。

インシデント：`ConnectionDown` インシデントが発生します。IF13 と IF31 からの `InterfaceDown` インシデントは `ConnectionDown` の下に相関付けされます。

ステータス：接続は危険な状態です。

結論：`ConnectionDown`

接続を操作できる



シナリオ：ページ 255 を参照してください。このシナリオは、*接続を操作できないシナリオ*に続いています。IF13 と IF31 の間の接続が現在稼働していると想定します。

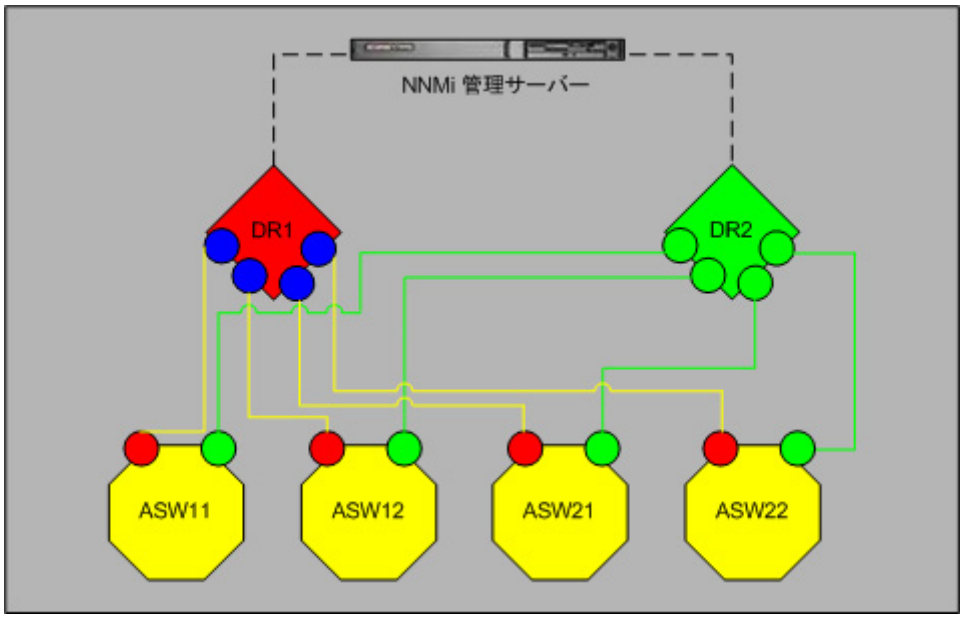
根本原因：IF13 と IF31 の間の接続が稼働しています。

インシデント：発生なし。`ConnectionDown` インシデントがクローズしました。

ステータス：接続は正常な状態です。

結論：`ConnectionUp`

直接接続しているノードが停止している



説明

DR1	分散ルーター 1
DR2	分散ルーター 2
ASW11	アクセススイッチ 11
ASW12	アクセススイッチ 12
ASW21	アクセススイッチ 21
ASW22	アクセススイッチ 22

シナリオ：アクセススイッチ ASW11、ASW12、ASW21、および ASW22 は、上で示すように分散ルーターに重複して接続されていると想定します。分散ルーター DR1 と DR2 は相互に直接接続しています。分散ルーター DR1 が停止します。

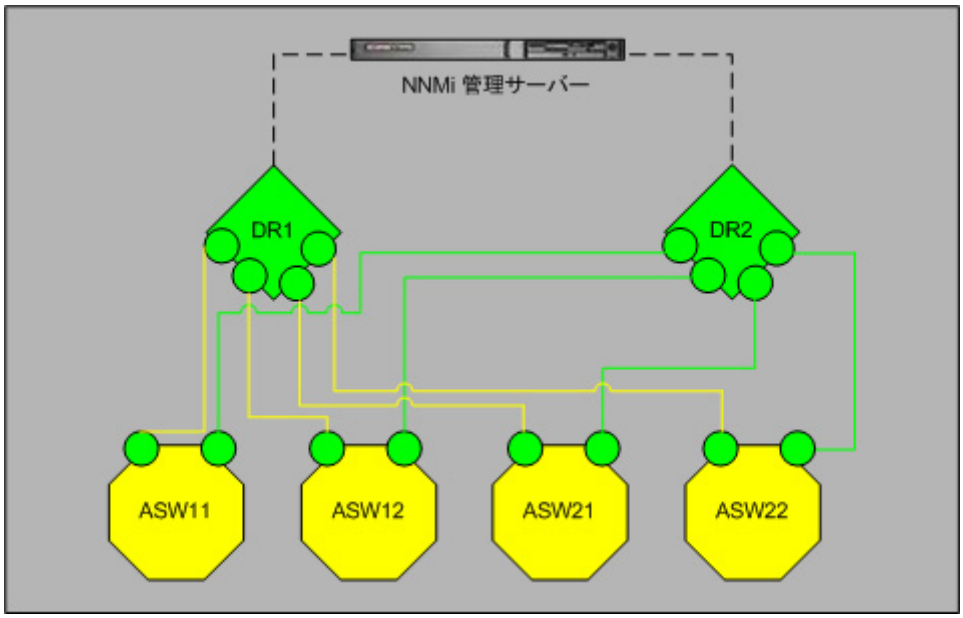
根本原因：ノード DR1 が隣接解析に従って停止しています。

インシデント：`NodeDown` インシデントが発生しました。1 ホップネイバーからの `InterfaceDown` インシデントが `NodeDown` インシデントの下に相関付けされます。

ステータス：ノードは危険な状態です。

結論：`NodeDown`

直接接続されたノードは稼働している



説明

DR1	分散ルーター 1
DR2	分散ルーター 2
ASW11	アクセススイッチ 11
ASW12	アクセススイッチ 12
ASW21	アクセススイッチ 21
ASW22	アクセススイッチ 22

シナリオ：ページ 257 を参照してください。このシナリオは、*直接接続されたノードは停止しているシナリオ*に続いています。分散ルーター DR1 が復帰していると想定します。

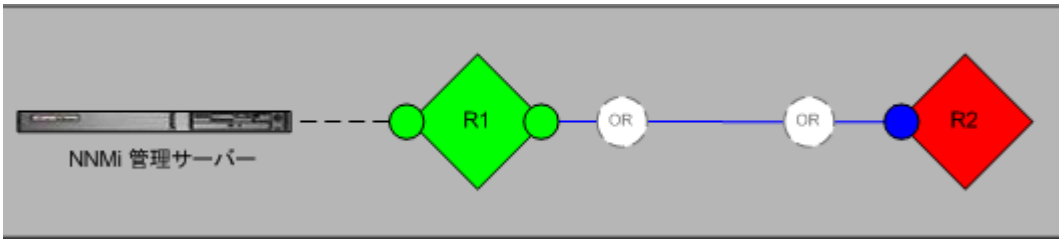
根本原因：ノード DR1 は稼働しています。

インシデント：発生なし。NodeDown インシデントが NodeUp インシデントによりクローズしています。

ステータス：ノードは正常な状態です。

結論：NodeUp

間接接続されたノードは停止している



説明

R1	ルーター 1
OR	光中継器（NNMi によって検出されていない）
R2	ルーター 2

上記の図表は概念的です。実際の NNMi トポロジマップまたはワークスペースビューを示していません。

シナリオ：このシナリオは、間接接続で NNMi が媒介デバイスを検出できない場合に発生します。この例では、ルーター R1 とルーター R2 は NNMi トポロジマップで直接接続しているように見えますが、実際は、これらの 2 つのルーターは光中継器経由で間接的に接続しています。（光中継器は SNMP または ICMP のクエリーに応答しないため、NNMi によって検出されません。）

ルーター R2 は到達不可能です。原因は、接続されたインタフェースが停止しているか、または光中継器との接続が切断されているかのいずれかです。間接的にルーター R2 に接続しているルーター R1 のインタフェースは、光中継器がまだ稼働中であるため、稼働中です。

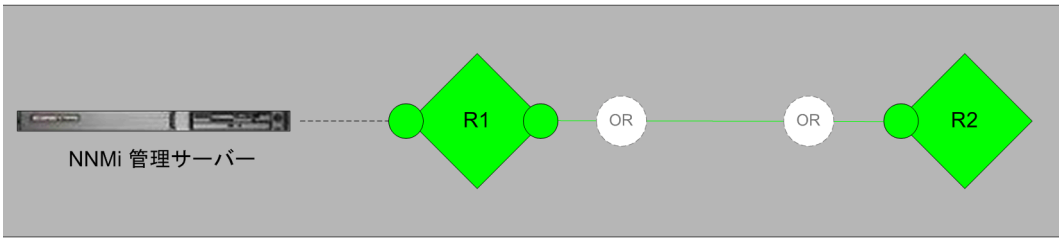
根本原因：ルーター R2 が隣接解析に従って停止しています。

インシデント：`NodeDown` インシデントが発生しました。

ステータス：ノード R2 は危険な状態です。

結論：`NodeDown`

間接接続されたノードは稼働している



説明

R1	ルーター 1
OR	光中継器（NNMi によって検出されていない）
R2	ルーター 2

上記の図表は概念的です。実際の NNMi トポロジマップまたはワークスペースビューを示していません。

シナリオ：このシナリオは、前の*間接接続されたノードは停止しているシナリオ*に続いています。失敗した接続がバックアップされて、ルーター R2 が到達可能になったと想定します。

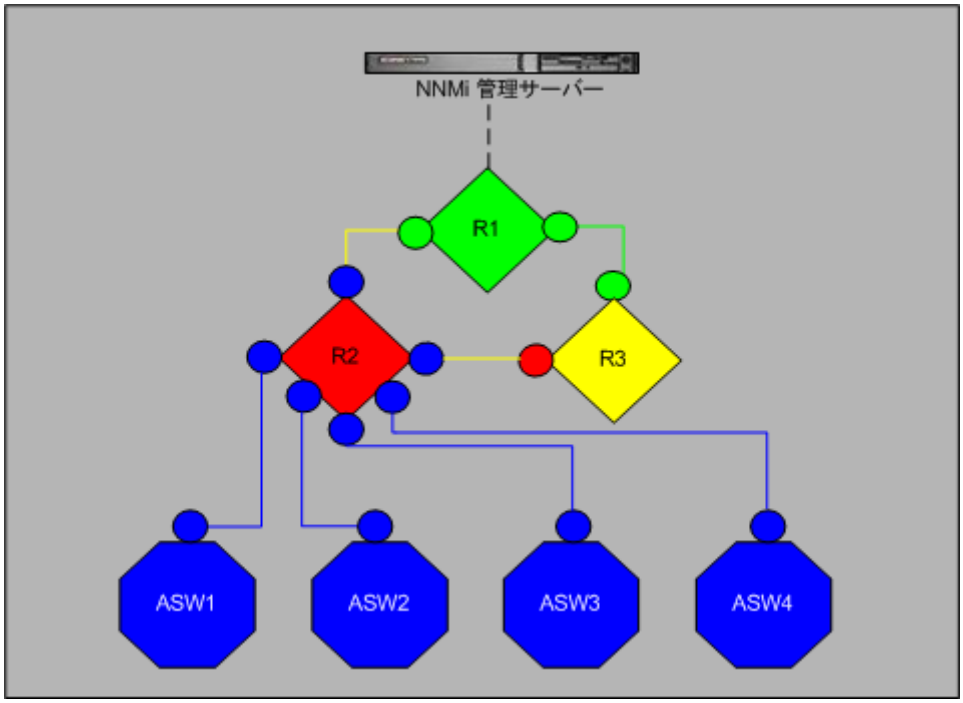
根本原因：R1 と R2 の間の接続が稼働しています。

インシデント：発生なし。`NodeDown` インシデントがクローズしました。

ステータス：ルーター R2 のステータスは正常域です。接続ステータスは正常域です。

結論：`NodeUp`

直接接続されたノードが停止しており、シャドウを作成する



説明

R1	ルーター 1
R2	ルーター 2
R3	ルーター 3
ASW1	アクセススイッチ 1
ASW2	アクセススイッチ 2
ASW3	アクセススイッチ 3
ASW4	アクセススイッチ 4

シナリオ：ルーター 2 (R2) が上で示すように停止します。

根本原因：ノード R2 が NNMi の隣接解析に従って停止しています。

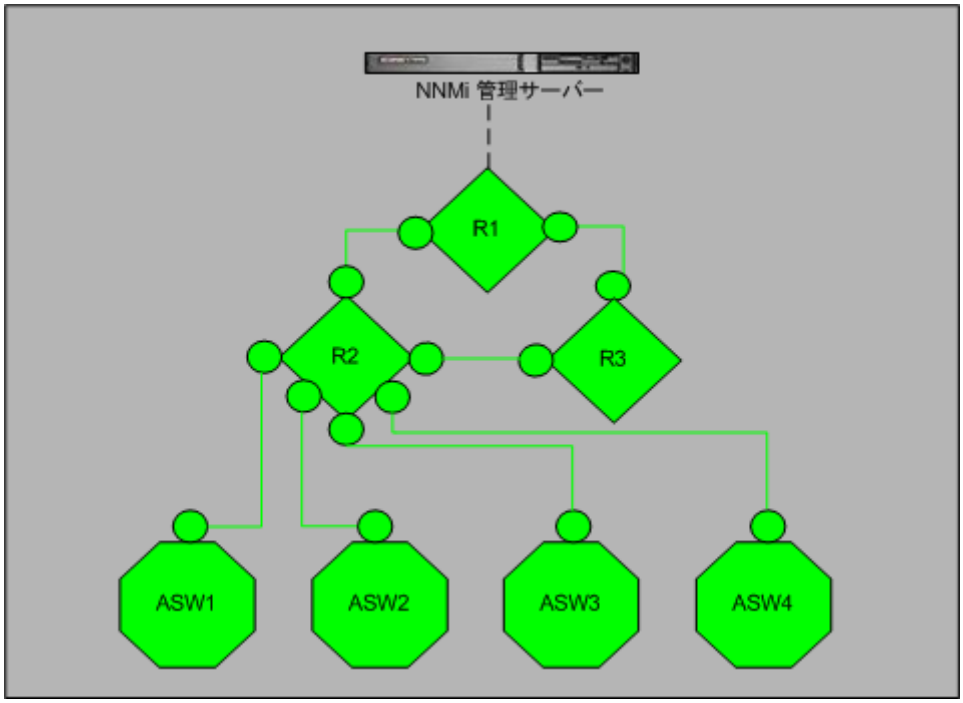
インシデント：`NodeDown` インシデントが発生しました。1 ホップネイバーからの `InterfaceDown` インシデントが `NodeDown` インシデントの下に相関付けされます。

ステータス：ノードは危険な状態です。

結論：`NodeDown`

結果：すべてのアクセススイッチが到達不可能です。シャドウ内のすべてのノードのステータスが不明であり、各ノードについての結論が `NodeUnmanageable` です。

直接接続されたノードが稼働しており、シャドウを除去している



説明

R1	ルーター 1
R2	ルーター 2
R3	ルーター 3
ASW1	アクセススイッチ 1
ASW2	アクセススイッチ 2
ASW3	アクセススイッチ 3
ASW4	アクセススイッチ 4

シナリオ：ページ 261 を参照してください。このシナリオは、*ノードが停止しており、シャドウを作成する*シナリオに続いています。R2 が上で示すように復帰していると想定します。

根本原因：ノード R2 は稼働しています。

インシデント：発生なし。NodeDown インシデントが NodeUp インシデントによりクローズしています。

ステータス：ノードは正常な状態です。

結論：NodeUp

結果：すべてのアクセススイッチが到達可能になっています。シャドウ内のすべてのノードのステータスは正常です。

重要ノードが到達不可能である

シナリオ：あるノードは重要ノードノードグループの一部ですが、このノードが到達不可能です。

ノードを重要ノードノードグループに、NmsApa サービスがこのノードを解析する前に追加する必要があります。ノードを重要ノードノードグループに追加する前に到達不可能になると、NmsApa サービスは NodeDown インシデントを発生しません。

根本原因：ノードは停止しています。NmsApa サービスは隣接解析を行いませんが、ノードが停止している理由は重要とマークされているためだけだと結論づけます。

インシデント：NodeDown インシデントが発生しました。相関インシデントは発生しません。

ステータス：ノードは危険な状態です。

結論：NodeDown

重要ノードが到達可能である

シナリオ：ページ 263 を参照してください。このシナリオは、*重要ノードが到達不可能である*シナリオに続いています。重要ノードが復帰しており、到達可能になったと想定します。

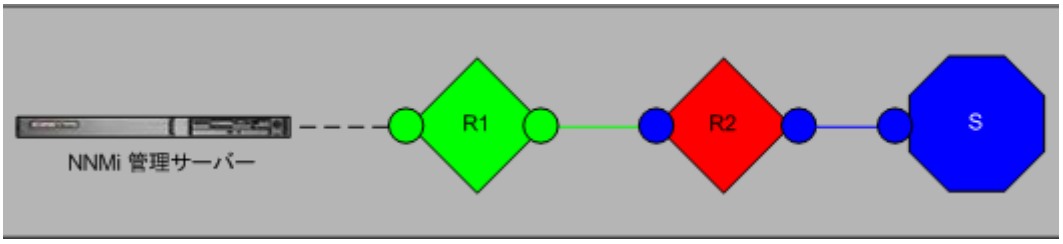
根本原因：ノードは稼働しています。

インシデント：発生なし。NodeDown インシデントが NodeUp インシデントによりクローズしています。

ステータス：ノードは正常な状態です。

結論：NodeUp

ノードまたは接続が停止している



説明

R1	ルーター 1
R2	ルーター 2
S	アクセススイッチ

シナリオ：ルーター 2 (R2) に対して冗長性がありません。R2 が停止しているか、ルーター 1 (R1) と R2 の間の接続が停止しています。

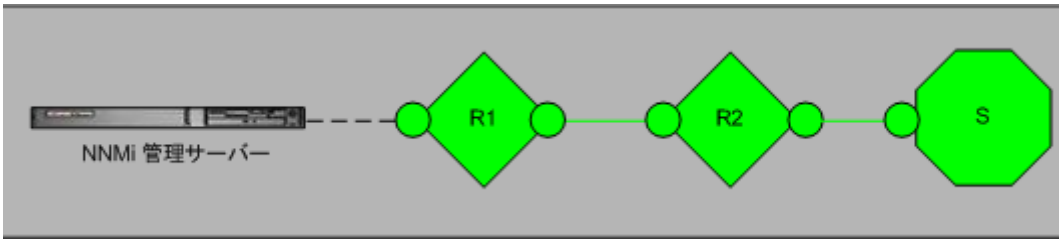
根本原因：ノードまたは接続は停止しています。

インシデント：`NodeOrConnectionDown` インシデントが発生しました。このシナリオのソースノードは R2 です。

ステータス：ノードは危険な状態です。接続は注意域の状態です。

結論：`NodeOrConnectionDown`

ノードまたは接続が稼働している



説明

R1	ルーター 1
R2	ルーター 2
S	アクセススイッチ

シナリオ：ページ 264 を参照してください。このシナリオは、*ノードまたは接続が停止しているシナリオ*に続いています。R2 が稼働状態になったと想定します。

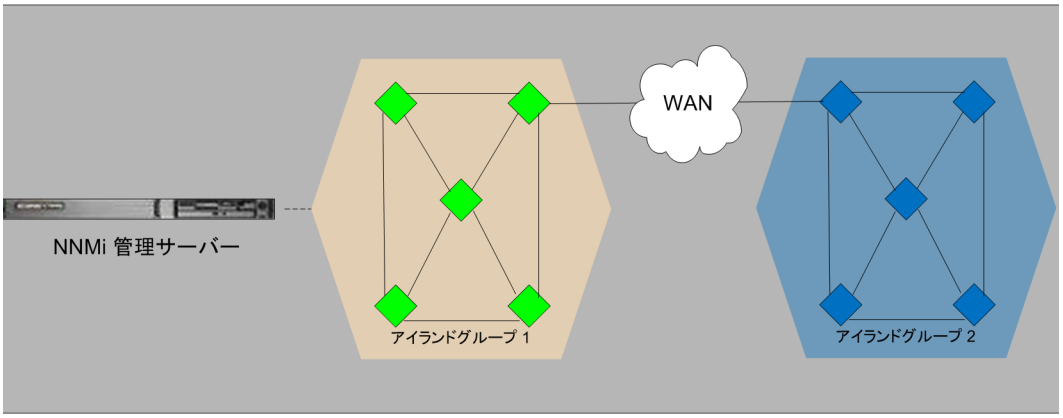
根本原因：NodeUp

インシデント：発生なし。`NodeOrConnectionDown` インシデントがクローズしました。

ステータス：ノードは正常な状態です。接続は正常な状態です。

結論：`NodeUp`

アイランドグループが停止している



上記の図表は概念的です。実際の NNMi トポロジマップまたはワークスペースビューを示していません。

シナリオ：NNMi はネットワークを 2 つのアイランドグループに分割しました。NNMi 管理サーバーは、アイランドグループ 1 のノードに接続されます。アイランドグループ 2 は、サービスプロバイダの WAN に問題が発生したため、到達不可能になっています。

アイランドグループには、その他のネットワークに接続されていないか、または最低限接続しているノードの高度に接続されたセットが含まれています。たとえば、NNMi は、WAN によって接続された地理的に分散されたサイトでエンタープライズネットワークの複数のアイランドグループを識別することができます。アイランドグループは NNMi によって作成され、ユーザーは変更することができません。アイランドグループに関する詳細については、NNMi コンソールヘルプを参照してください。

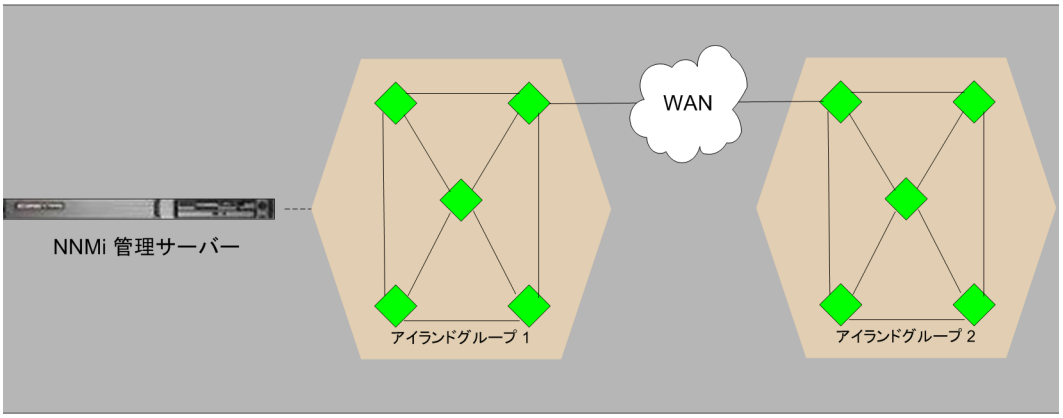
根本原因：アイランドグループ 2 が隣接解析に従って停止しています。

インシデント：`IslandGroupDown` インシデントが発生しました。NNMi はインシデントのソースノードとしてアイランドグループ 2 から代表ノードを使用します。

ステータス：アイランドグループ 2 のステータスは［不明］に設定されています。アイランドグループ 2 のオブジェクトは不明ステータスを持っています。アイランドグループ 1 の接続インタフェースは、稼働 WAN への接続がまだ稼働しているため、稼働しています。

結論：アイランドグループへの適用不可

アイランドグループが稼働している



上記の図表は概念的です。実際の NNMi トポロジマップまたはワークスペースビューを示していません。

シナリオ：このシナリオは、前の*アイランドグループは停止しているシナリオ*に続いています。サービスプロバイダの WAN 問題が修正され、アイランドグループ 2 が到達可能になったと想定します。

根本原因：アイランドグループ 2 への WAN 接続はバックアップです。

インシデント：発生なし。`IslandGroupDown` インシデントがクローズしました。

ステータス：アイランドグループ 2 のステータスは［正常域］に設定されています。アイランドグループ 2 のオブジェクトは正常域ステータスに戻ります。

結論：アイランドグループへの適用不可

ネットワーク設定の変更

1 日のうちで、NNMi オペレータはいくつかの設定変更を行うことがあります。以下のシナリオは、いくつかの共通ネットワーク設定変更について説明し、NNMi がこれらの変更に対してどう対応するかを示しています。

ノード更新中

ネットワークオペレータが、たとえば故障したインタフェースボードを正常な代替品と交換して、ノードを変更する場合を想定します。NNMi がこの変更を認識すると、検出プロセスは NmsApa サービスに通知を送信します。NmsApa サービスはこの通知を使用して、次のタスクを完了します。

- ノードのステータスを再計算します。
- ノード上の削除した IPv4 アドレスおよびインタフェースのすべての登録済インシデントをクローズします。

インタフェースが接続に加入および離脱する

ネットワークオペレータがネットワークデバイスを接続する方法を変更する場合を想定します。インタフェースが接続に加入したり 1 つの接続を離れて別の接続に加入すると、NNMi 検出プロセスは NmsApa サービスに通知を送信します。NmsApa サービスはこの通知を使用して、接続のステータスを再計算します。

デバイスがトラップを発生した場合

ColdStart トラップと WarmStart トラップ — NmsApa サービスは、ColdStart トラップと WarmStart トラップのイベントシステムからの通知を登録します。これらの通知が行われると、NmsApa サービスはそのトラップを発生したノードからのデバイス情報の再検出を開始します。

LinkUp トラップと LinkDown トラップ — NmsApa サービスは、LinkUp トラップおよび LinkDown トラップのイベントシステムからだけでなく、受信するいくつかのベンダ固有のリンクトラップからの通知も登録します。これらの通知が行われると、NmsApa サービスはそのトラップを発生したノードからのデバイス情報の再検出を開始します。

NNMi が提供するトラップインシデント設定の一覧は、NNMi ヘルプを参照するか、［インシデントの設定］ビューで［**SNMP トラップの設定**］タブを選択してください。

NNMi 管理設定の変更

1 日のうちで、NNMi ツール管理者はいくつかの NNMi 設定変更を行うことがあります。次のシナリオは、いくつかの共通 NNMi 管理設定変更を説明し、NNMi がこれらの変更に対してどう対応するかを示しています。

- **NNMi 管理者は IPv4 アドレスの管理を解除するか、サービス停止にする**

  NmsApa サービスが、StatePoller からの通知を、pingState がポーリングなしに設定された後で受け取ります。NmsApa サービスはこの通知に反応して、IPv4 アドレスのステータスをステータスなしに設定します。

- **NNMi 管理者は IPv4 アドレスを管理するか、サービス状態に戻す**

  NmsApa サービスが、StatePoller からの通知を、pingState が測定された値に設定された後で受け取ります。NmsApa サービスはこの通知に反応して、IPv4 アドレスのステータスを、測定された値に基づいて計算します。

- **NNMi 管理者はインタフェースの管理を解除するか、サービス停止にする**

  NmsApa サービスが、StatePoller からの通知を、operState がポーリングなしに設定された後で受け取ります。NmsApa サービスはこの通知に反応して、インタフェースのステータスをステータスなしに設定します。

- **NNMi 管理者はインタフェースを管理するか、サービス状態に戻す**

  NmsApa サービスが、StatePoller からの通知を、operState が測定された値に設定された後で受け取ります。NmsApa サービスはこの通知に反応して、インタフェースのステータスを、測定された値に基づいて計算します。

- **NNMi 管理者はノードの管理を解除するか、サービス停止にする**

  NmsApa サービスが、StatePoller からの通知を、agentState がポーリングなしに設定された後で受け取ります。すべてのインタフェースで operState がポーリングなしに設定され、すべての IPv4 アドレスで pingState がポーリングなしに設定されます。NmsApa サービスはこの通知に反応して、ノードのステータスをステータスなしに設定します。

- **NNMi 管理者はノードを管理するか、サービス状態に戻す**

  NmsApa サービスが、StatePoller からの通知を、agentState が測定された値に設定された後で受け取ります。すべてのインタフェースで operState が測定された値に設定され、すべての IPv4 アドレスで pingState が測定された値に設定されます。NmsApa サービスはこの通知に反応して、ノードのステータスを計算します。

用語集

A

ARP cache

ARP（アドレス解決プロトコル）キャッシュは、データリンク層（OSI レイヤー 2）アドレスをネットワーク層（OSI レイヤー 3）アドレスにマップするオペレーティングシステムテーブルです。データリンク層アドレスは通常は MAC アドレスですが、ネットワーク層アドレスは通常は IP アドレスです。ルールベースの検出では、NNMi は、検出されたノードで ARP キャッシュエントリ（ならびに他のテクニック）を使って、現在の検出ルールに照らしてチェックできる追加ノードを見つけます。

C

Causal Engine

因果関係ベースの方法を使って、根本原因解析 (RCA) をネットワーク現象に適用する NNMi テクノロジ。Causal Engine RCA をトリガーするのは、状態ポーリング、SNMP トラップ、特定のインシデントの結果として検出された変更など、特定のオカレンスです。Causal Engine は RCA を使って、管理対象オブジェクトの status を調べ、これらオブジェクトに関する conclusion を明確化し、根本原因インシデントを生成します。

conclusion

NNMi で、管理対象オブジェクト用に Causal Engine が status と根本原因インシデントを決定した方法を明らかにする Causal Engine が生成および使用するサポート詳細。

H

HA

「高可用性」を参照。

HA リソースグループ

HP ServiceGuard、Veritas Cluster Server、Microsoft Cluster Service などの最新の高可用性環境では、アプリケーションは、アプリケーション自体、その共有ファイルシステム、仮想 IP アドレスのようなリソースの復合物として表わされます。リソースは *HA リソースグループ*で構成されます。これはクラスタ環境で実行中のアプリケーションを表します。

HP Network Node Manager i Software

ネットワーク管理の支援や統合のために設計された HP のソフトウェア商品です（短縮形は NNMi）。ネットワークノードの継続検出、イベントの監視、ネットワーク障害管理といった機能を備えています。おもに NNMi コンソールからアクセスします。

HP Network Node Manager i-series Software

HP NNM i-series Softwareとも呼ばれます。HP Network Node Manager i Software 製品 (NNMi) の周りに構築された HP ソフトウェア製品のファミリです。HP NNM i-series Software には、NNMi、NNMi Advanced、NNM iSPI for Performance、NNM iSPI for MPLS のような関連 iSPI が含まれます。

I

ICMP

インターネット制御メッセージプロトコルを参照してください。

interface group

NNMi の主要なフィルタテクニックの 1 つ。ただし、グループごとに、グループまたはフィルタ視覚化に設定を適用する目的で、インタフェースはグループにまとめられます。インタフェースグループは、モニタリングの設定、テーブルビューのフィルタ、マップビューのカスタマイズのいずれか、またはすべてに使用できます。「ノードグループ」も参照。

iSPI

「NNM iSPI」を参照。

J

jboss アプリケーションサーバー

NNMi プロセスは、ovjboss と呼ばれる jboss アプリケーションサーバーを利用します。このアプリケーションサーバーは主要な NNMi プロセスであり、Java 2 Platform、Enterprise Edition (J2EE)、Enterprise Java Beans (EJB) すべてのビジネスロジックとデー

タベースアクセス機能をホストします。NNM iSPI には独自の jboss アプリケーションサーバーがあります。

JGroups

マルチキャスト通信およびクラスタリング用のオープンソース技術で、jboss がクラスタリング機能のために内部で使用します。

L

L2

レイヤー 2 を参照してください。

L3

レイヤー 3 を参照してください。

M

MIB

管理情報ベースを参照してください。

N

NNM 6.x/7.x イベント

古い NNM 管理ステーションから NNMi に転送されたイベント用の NNMi 用語。NNMi には、転送されたイベントから NNMi が生成するインシデントを参照するためのインシデントビューがあります。

NNM iSPI

HP Network Node Manager i-series Software ファミリ内のスマートプラグイン。NNM iSPI は、MPLS のような特殊テクノロジ用に、またはネットワークエンジニアリングのような特定の分野用に、NNMi に機能を追加します。

NNMi

HP Network Node Manager i Software を参照してください。

NNMi コンソール

NNMi ユーザーインタフェース。オペレータや管理者は、NNMi コンソールを使って NNMi ネットワーク管理タスクを実行できます。

O

OID

オブジェクト識別子を参照してください。

ovstatus コマンド

NNMi が管理するプロセスの現在のステータスを報告するコマンドです。NNMi コンソール（**[ツール] [NNM ステータス]**）またはコマンドプロンプトで起動できます。*ovstatus* リファレンスページまたは UNIX のマンページを参照してください。

ovstart コマンド

NNMi の管理プロセスを起動するためのコマンドです。コマンドプロンプトで起動します。*ovstart* リファレンスページまたは UNIX のマンページを参照してください。

ovstop コマンド

NNMi の管理プロセスを停止するためのコマンドです。コマンドプロンプトで起動します。*ovstop* リファレンスページまたは UNIX のマンページを参照してください。

P

ping スイープ

ICMP ECHO 要求を複数の IP アドレスに送信し、応答するノードにどのアドレスが割り当てられているか調べるネットワークプローブテクニック。ルールベースの検出で有効化されている場合、NNMi は、設定 IP アドレス範囲について ping スイープを使用し、追加ノードを見つけることができます。サービス拒絶攻撃に ping スイープを使用できるので、ICMP ECHO 要求をブロックするネットワーク管理者もいます。

PostgreSQL

トポロジ、インシデント、設定情報のような情報を保存するために NNMi がデフォルトで使用するオープンソースリレーショナルデータベース。NNMi では、ほとんどのテーブルについて PostgreSQL の代わりに Oracle を使用するよう設定することもできます。

R

RCA

根本原因解析を参照してください。

S

SNMP

簡易ネットワーク管理プロトコル (SNMP) を参照してください。

SNMP トラップ

ポーリングによるネットワーク管理（SNMP エージェントから請求された応答）は、処理をできるだけ簡単にするための SNMP の設計原則です。しかし、このプロトコルは、SNMP エージェントから SNMP マネージャプロセス（この場合、NNMi）への要請されないメッセージの通信も提供します。要請されないエージェントメッセージは、「トラップ」として知られており、内部状態の変化または障害条件に応答して SNMP

エージェントが生成します。NNMi は、受信した SNMP トラップ（**[SNMP トラップ]** インシデントの参照ビューに表示）からインシデントを生成します。

SNMP トラップストーム

要請されない大量の SNMP エージェントメッセージ。SNMP マネージャプロセス（この場合、NNMi）を圧倒する可能性があります。`nnmtrapconfig.ovpl` スクリプトを使用して SNMP トラップストームしきい値を NNMi に指定できます。受信トラップレートが指定のしきい値レートを超えるとき、NNMi は、トラップレートが再対応レート未満に下がるまでトラップをブロックします。

status

NNMi では、全般的な稼働状態を示す管理対象オブジェクトの属性。ステータスは、管理対象オブジェクトの未解決 conclusion から Causal Engine が計算します。状態と比べてください。

sysObjectID

システムオブジェクト ID を参照してください。

あ

アカウント

「ユーザーアカウント」を参照。

アクティブなクラスタノード

「アクティブなサーバー」を参照。

アクティブなサーバー

高可用性設定で NNMi プロセスを現在実行しているサーバー。

アドレスのヒント

「検出のヒント」を参照。

い

因果関係

あるイベント（原因）と別のイベント（影響）の間の関係を示します。イベント（影響）は最初のイベント（原因）の直接的な結果です。NNMi は、因果関係分析アルゴリズムを使用して、イベントのサイクルを分析し、ネットワーク問題を解決するソリューションを明らかにします。

インシデント

NNMi では、ネットワークに関連するオカレンスの通知は、NNMi コンソールインシデントビューとフォームに表示されます。NNMi には、インシデント属性に基づいてユーザーがインシデントをフィルタできるようにするいくつもの **[インシデントの管理]** ビューと **[インシデントの参照]** ビューがあります。ほとんどのインシデントビューには、NNMi（*管理イベント*と呼ばれることもあります）が直接生成したインシデントが表示されます。NNMi には、SNMP トラップから生成されたインシデントおよび NNM 6.x/7.x イベントから生成されたインシデントを参照するビューもあります。

インターネット制御メッセージプロトコル

中核的なインターネットプロトコルスイート (TCP/IP) の 1 つ。ICMP ping は、状態ポーリング用の SNMP クエリーとともに NNMi が使います。

インタフェース

ノードをネットワークに接続するのに使われる物理ポート。

え

エピソード

NNMi 根本原因解析で、特定の持続時間を指すのに使う用語。この持続時間は一次的な障害によってトリガーされ、その間、二次障害は抑制されるか、または一次的障害の下で相互に関連付けられます。

お

オブジェクト識別子

SNMP で、MIB データオブジェクトを識別する数字のシーケンス。OID は、小数点で分離された数字で構成されます。各数字は、MIB 階層のそのレベルにおける特定のデータオブジェクトを表します。OID は MIB オブジェクト名と同等の数字です。たとえば、MIB オブジェクト名 `iso.org.dod.internet.mgmt.mib-2.bgp.bgpTraps.bgpEstablished` はその OID `1.3.6.1.2.1.15.0.1` と同等です。

か

仮想 IP アドレス

特別なネットワークハードウェアに結び付かれていない IP アドレス。現在のフェイルオーバーまたはロードバランシングのニーズに基づいて、最も該当するサーバーに中断されないネットワークトラフィックを送信するため、高可用性設定で使われます。

仮想ホスト名

仮想 IP アドレスと関連付けられたホスト名。

簡易ネットワーク管理プロトコル (SNMP)

OSI モデルのアプリケーション層（レイヤー 7）で機能する簡易なプロトコル。リモートユーザーは、このプロトコルによって、ネットワーク要素の管理情報を検査または変更できます。SNMP は、管理対照ノード上のエージェントプロセッサとネットワーク管理情報

を交換するために NNMi が使う主要なプロトコルです。NNMi は、次の 3 つの SNMP の最も一般的なバージョンをサポートしています。

管理サーバー

NNMi 管理サーバーは、NNMi ソフトウェアがインストールされるコンピュータシステムです。NNMi のプロセスとサービスは、NNMi 管理サーバーで稼働します。(以前の NNM リビジョンはこのシステムについて「NNM 管理ステーション」という用語を使用していました)

管理情報ベース

SNMP で、管理対照ネットワークに関するデータの階層的に組織化された集合。管理情報ベース内のデータオブジェクトは管理対照デバイスの特色を参照します。NNMi は、ネットワーク管理情報を収集する場合、MIB データオブジェクト (「MIB オブジェクト」、「オブジェクト」、「MIB」と呼ばれることもあります) を使って、管理対照ノードとの間で SNMP クエリーを出し、または SNMP トラップを受け取ります。

く

組み込みデータベース

NNMi に組み込まれたデータベース。NNMi は、ほとんどのテーブルについて、組み込みデータベースの代わりに外部の Oracle データベースを使うよう設定することもできます。「PostgreSQL」も参照。

クラスタ

NNMi の関係では、高可用性テクノロジまたは jboss クラスタ化機能によってリンクされるハードウェアおよびソフトウェアのグループ化のことです。これらは、一緒に機能して、コンポーネントに過剰負荷または障害が発生した場合、機能とデータの連続性を確実にします。クラスタ内のコンピュータは一般に高速 LAN 経由でお互いに接続されます。クラスタは、通常、可用性またはパフォーマンスを向上させるために導入します。

クラスタメンバーまたはノード

NNMi の関係では、NNMi 高可用性をサポートするよう設定された、または設定される予定の高可用性クラスタ内のシステム。

け

検出シード

「シード」を参照。

検出のヒント

SNMP ARP キャッシュクエリー、CDP、EDP、またはその他の検出プロトコルクエリー、または ping スイープによって NNMi が見つけた IP アドレス。NNMi はさらに、検出ヒントとして見つかった IP アドレスについてクエリーを実行し、結果をルールベースの検出内の現在の検出ルールに照らしてチェックします。

検出プロセス

NNMi が、ネットワークノードを管理下におくために、これらの情報を収集するプロセス。初期検出は、まずデバイスインベントリの情報を収集し、次にネットワーク接続情報を収集するという 2 つのフェーズのプロセスで実行されます。

最初の検出の後も検出プロセスは継続されます。つまり、リストに基づいた検出では、シードリスト内のデバイスは、設定が変更されると更新されます。ルールベースの検出では、新しいデバイスは現在の検出ルールに合致すると追加されます。検出プロセスは、NNMi コンソールまたはコマンドラインから、デバイスまたはデバイスセットについてオンデマンドで開始できます。

「スパイラル検出」、「ルールベースの検出」、「リストに基づいた検出」も参照してください。

検出ルール

ルールベースの検出プロセスを制限するのに使われる、ある範囲のユーザー定義 IP アドレスまたはシステムオブジェクト ID (OID)。検出ルールは、NNMi コンソールの **[自動検出ルール]** の **[検出の設定]** 部分に設定します。「ルールベースの検出」も参照。

こ

高可用性

このガイドでは、設定の一部に障害があっても中断されないサービスを提供するハードウェアおよびソフトウェアの設定のことです。高可用性 (HA) というのは、コンポーネントに障害があった場合でもアプリケーションを実行し続けるよう冗長コンポーネントを備えた構成のことです。NNMi は、いくつかの市販の HA ソリューションのうちの 1 つをサポートするよう構成できます。

コミュニティ文字列

SNMP クエリーを SNMP エージェントに認証するために、SNMPv1 および SNMPv2C システムで使われるパスワード類似のメカニズム。コミュニティ文字列は SNMP パケット内のクリアテキストに渡されるので、パケット傍受に対して脆くなります。SNMPv3 は、認証用の強力なセキュリティメカニズムを用意します。

コンソール

「NNMi コンソール」を参照。

し

シード

ネットワーク検出プロセスの開始点として機能することによって、NNMi のネットワーク検出を補助するネットワークノードのことです。たとえば、管理環境内のコアルーターなどがシードになることができます。各シードは、IP アドレスやホスト名によって識別されます。ルールベースの検出が設定されていない場合、NNMi の検出プロセスは指定シードのリストに基づいた検出に制限されます。

シード検出

リストに基づいた検出を参照してください。

システムアカウント

NNMi では、NNMi のインストール時に使うために備わっている特別なアカウントです。NNMi システムアカウントは、インストール終了後は、コマンドラインのセキュリティや復旧目的のみに使用されます。ユーザーアカウントと比べてください。

システムオブジェクト ID

NNMi で、ネットワーク要素のモデルまたは種類を識別する SNMP オブジェクト識別子の専門化された用語。システムオブジェクト ID は、ネットワーク要素の MIB オブジェクトの一部です。このオブジェクトは、検出の間に個別のノードから NNMi がクエリーします。システムオブジェクト ID によって分類できるネットワーク要素の種類の例には、HP ProCurve スイッチファミリ、HP J8715A ProCurve Switch、HP IPF システム用の HP SNMP エージェントがあります。他のベンダーのネットワーク要素も同じようにシステムオブジェクト ID に従って分類できます。システムオブジェクト ID の重要な使用法は NNMi デバイスプロファイルの定義にあります。デバイスプロファイルは、ネットワーク要素の種類が分かると、削減できるネットワーク要素の特徴を指定します。

自動検出

「ルールベースの検出」を参照。

障害ポーリング

主要な NNMi モニタリングアクティビティ。このアクティビティでは、NNMi は、管理対象の各オブジェクトの状態を調べるために、管理対象インタフェース、IP アドレス、SNMP エージェントすべてに関し、ステータス MIB の SNMP 読み取り専用クエリーまたは ICMP ping を発行します。ユーザーは、NNMi コンソールの **[設定]** ワークスペースの **[モニタリングの設定]** で、さまざまなインタフェースグループ、ノードグループ、ノードすべてについて実行された障害ポーリングの種類をカスタマイズできます。障害ポーリングは状態ポーリングのサブセットです。

状態

NNMi では、一般的に、MIB II `ifAdminStatus`、MIB II `ifOperStatus`、パフォーマンス、または可用性に関連する自己報告された管理対象オブジェクト応答について**状態**という用語を使用します。status と比べてください。

状態ポーリング

NNMi の State Poller が実行する指令されたモニタリング。障害、パフォーマンス、コンポーネント稼働状態、管理対象オブジェクトの可用性データを取得するために ICMP ping と SNMP クエリーを使います。「障害ポーリング」も参照。

す

スパイラル検出

NNMi の管理するネットワークのインベントリ、コンテインメント、リレーションシップ、接続についての情報などのネットワークトポロジ情報を NNMi が常時更新する処理のことです。「検出プロセス」、「ルールベースの検出」、「リストに基づいた検出」も参照してください。

と

トポロジ（ネットワーク）

ネットワークのノードや接続などが、通信ネットワーク上でどのように配置されているのかを示す図のことです。

トラップ

SNMP トラップを参照してください。

ね

根本原因インシデント

*Correlation Nature（相関関係の性質）*属性が *Root Cause（根本原因）*に設定されている NNMi インシデント。NNMi は、関連問題の現象が処理されていない場合、根本原因解析 (RCA) を使って現象を解決するすぐ実施できる課題として根本原因インシデントを確定します。「根本原因解析」を参照。

根本原因解析

NNMi で、根本原因解析 (RCA) とは、ネットワーク問題の原因を調べるために NNMi が使う問題解決方法のクラスのことです。NNMi で、根本原因とは、関連付けられた問題の現象が処理されていない場合、すぐに実施できる問題です。NNMi は、次の 2 つの主要な方法で根本原因の識別を使います。根本原因が解決さ

れるまで、すぐに実施できる問題についてユーザーに通知し、二次的問題の現象を報告しないようにします。根本原因の判定の結果、管理対象オブジェクトのステータス変更または根本原因インシデントの生成が行われることがあります。

NNMi が RCA を使う方法の例を挙げると、管理対象ルーターに障害が発生し、NNMi 管理サーバーからルーターの別の側にある管理対象ノードが状態ポーリングクエリーに応答できなくなるというシナリオです。NNMi は RCA を使用し、状態ポーリング障害が二次的問題の現象であるか調べます。ルーターが根本原因インシデントであることを報告し、根本原因ルーター障害が解決されるまでダウンストリームノードで発生している問題の現象を報告することは差し控えます。

の

ノード

ネットワーク関係で、ネットワークに接続されているコンピュータシステムやデバイス（プリンタ、ルーター、ブリッジなど）のことです。SNMP クエリーに応答できるノードは最も包括的な情報を NNMi に提供しますが、NNMi は非 SNMP ノードの制限された管理も実行できます。

ノードグループ

NNMi の主要なフィルタテクニックの 1 つ。ただし、グループごとに、グループまたはフィルタの視覚化に設定を適用するため、ノードはグループにまとめられます。ノードグループは、モニタリングの設定、テーブルビューのフィルタ、マップビューのカスタマイズのいずれか、またはすべてに使用できます。「interface group」も参照。

は

パブリックキー証明書

ネットワークセキュリティおよび暗号化で使用されます。デジタル署名を組み込み、パブリックキーと識別情報を結合するファイルです。証明書は、パブリックキーが個人または組織に属することの確認に使われます。NNMi は SSL 証明書を使います。これにはクライアント / サーバー通信の認証と暗号化のために、パブリックキーおよびプライベートキーが含まれています。

ほ

ポート

ネットワークハードウェアの関係において、ネットワークデバイスを経由して情報の受け渡しを行うコネクタです。

ボリュームグループ

コンピュータストレージ仮想化の用語。1 つの大規模ストレージエリアを形成するよう設定された 1 つまたは複数のディスクドライブ。NNMi がサポートするいくつかの高可用性製品は、共有ファイルシステムにおいてボリュームグループを使用します。

み

未接続インタフェース

NNMi の観点からは、未接続インタフェースは NNMi が検出した他のデバイスに接続されていないインタフェースのことです。デフォルトでは、NNMi がモニタリングする未接続インタフェースは IP アドレスのあるもののみであり、**[ルーター]** ノードグループのノードに含まれます。

ゆ

ユーザーアカウント

NNMi では、ユーザーまたはユーザーグループのために NNMi にアクセスする方法を提供します。NNMi ユーザーアカウントは NNMi コンソールにセットアップされ、事前定義されたユーザーロールを実装します。「システムアカウント」および「ユーザーロール」を参照。

ユーザーロール

NNMi 管理者は、ユーザーアクセス設定の一環として、NNMi の各ユーザーアカウントに定義済みのユーザーロールを割り当てます。ユーザーロールとは、どのユーザーが NNMi コンソールにアクセスができるのか、またどのワークスペースとアクションを各ユーザーが利用できるかを決定します。NNMi には、以下の階層的ユーザーロールがあります。これらはプログラムによってあらかじめ定義されており変更できません。*管理者*、*Web サービスクライアント*、*オペレータレベル 2*、*オペレータレベル 1*、*ゲスト*。「ユーザーアカウント」も参照。

り

リストに基づいた検出

シードのリストに基づいたプロセス。*シードとして指定するノードのみに関する*詳細ネットワーク情報を検出し、返します。リストに基づいた検出は、特定したクエリーとタスクのネットワークインベントリのみを保守します。ルールベースの検出と比べてください。「検出プロセス」と「スパイラル検出」も参照。

領域

NNMi において、タイムアウト値やアクセス資格認定のような通信設定を行うためにグループにまとめられたデバイス。

る

ルール

「検出ルール」を参照。

ルールベースの検出

*自動検出*と呼ばれることがよくあります。NNMi は、ルールベースの検出を使い、ユーザー指定検出ルールに従って、NNMi がデータベースに追加する必要のあるノードを探し出します。NNMi は、検出されたノードのデータ内で検出のヒントを探してから、指定の検出ルールに照らしてこれら候補をチェックします。検出ルールは、NNMi コンソールの［**自動検出ルール**］の［**検出の設定**］部分に設定します。リストに基づいた検出と比べてください。

れ

レイヤー 2

階層化通信モデルである Open Systems Interconnection (OSI) のデータリンク層です。データリンク層では、ネットワークの物理リンクを介してデータの伝送を行います。NNMi レイヤー 2 ビューは、デバイスの物理接続に関する情報を提供します。

レイヤー 3

階層化通信モデルである Open Systems Interconnection (OSI) のネットワーク層です。ネットワーク層は、ネットワーク上の隣接するノードのアドレスの取得、データ伝送経路の選択、サービス品質などに関与します。NNMi レイヤー 3 ビューは、ルーティングの観点から接続に関する情報を提供します。

ろ

ロール

「ユーザーロール」を参照。

論理ボリューム

個別のファイルシステムまたはデバイススワップ空間として使えるボリュームグループ内の任意のサイズの容量を指すコンピュータストレージ仮想化の用語。NNMi がサポートするいくつかの高可用性製品は共有ファイルシステムで論理ボリュームを使います。

索引

記号

<resource_group>.cntl.log ファイル
    HP-UX, 120
    Linux, 120
/etc/hosts, 100

A

AddressNotResponding インシデント , 246
AlarmPoint
    Java Client, 223
    エージェント , 223
    クライアント , 223
    サーバー , 223
    サポートされるバージョン , 222
    統合
        概要 , 221 ～ 223
        使用法 , 223
        トラブルシューティング , 224
        無効化 , 224
        有効化 , 223
    ドキュメント , 223
APAgent (AlarmPoint), 223
APClient (AlarmPoint), 223
Application_A.log ファイル , 121
ARP キャッシュ , 51

B

BIND, 100

C

Causal Engine
    概念 , 242
    機能 , 241
    ネットワーク失敗のシナリオ , 246
CIA, 179
Cisco
    スイッチ
        階層 , 31
        ノードグループの定義 , 30
    ルーター
        階層 , 31
        ノードグループの定義 , 30
Client、AlarmPoint Java, 223
cluster.exe コマンド , 97
cluster.log ファイル , 120
cmcluster ファイル , 120
ConnectionDown インシデント , 246
ConnectionPartiallyUnresponsive インシデント , 246
CPU リリース , 69

D

DiskGroup_A.log ファイル , 121
DNS, 100

E

Event Configurator ウィンドウ、
    NNM 6.x/7.x, 175, 183

F

FORWARD フィールド、NNM 6.x/7.x イベント
    設定 , 176

G

GUI。NNMi コンソール*を参照*

H

HA_nnmhaserver.log ファイル
    UNIX, 120
    Windows, 120
haconfigure.log ファイル
    UNIX, 120
    Windows, 120

HA クラスタ
IP アドレスの変更
NNMi, 99 ～ 102
NNM iSPI for Performance, 103
アーキテクチャ , 79
概念 , 79
起動の問題
nmsdbmgr, 116
NNMi, 115
pmd, 116
共有データ , 96 ～ 97
サポート対象の製品 , 78
シナリオ , 80
設定
NNMi, 84 ～ 88
NNM iSPI for Performance, 91 ～ 95
アドオン iSPI, 89
情報、NNMi, 84
情報、NNM iSPI for Performance, 91
スクリプト , 118
ファイル , 118
マンページ , 83
リファレンスページ , 83
ログファイル , 120
設定解除
NNMi, 104 ～ 107
NNM iSPI for Performance, 107 ～ 109
設定のトラブルシューティング , 113 ～ 117
前提条件 , 78
ホスト名の変更
NNMi, 99 ～ 102
NNM iSPI for Performance, 103
メンテナンス
NNMi, 99 ～ 102
NNM iSPI for Performance, 102 ～ 103
アドオン iSPI, 102
メンテナンスモード , 99
用語 , 80
ライセンス , 98
HA クラスタの設定解除
NNMi, 104 ～ 107
NNM iSPI for Performance, 107 ～ 109
HA クラスタ用の NNMi を再び有効にする , 115
HA 設定 , 118
HA 設定のメンテナンス , 99 ～ 103
HA 用のクラスタアーキテクチャ , 79
HA 用のメンテナンスモード , 99
HA リソースグループ
起動できない , 114
設定
NNMi, 84
NNM iSPI for Performance, 92
説明 , 80
停止
NNMi, 105
NNM iSPI for Performance, 108
hostname コマンド , 100
［HP Network Node Manager i-series］ウィンドウについて , 68
HPOM 統合
値 , 188
概要 , 187
サポートされるバージョン , 189
使用法 , 193
トラブルシューティング , 195 ～ 197
ドキュメント , 189
無効化 , 194
有効化
UNIX, 191
Windows, 190
HPOM 統合の無効化 , 194
HPOM へのインシデント転送 , 188
HP Operations Manager
HP Performance Insight, 23
HP ServiceGuard
HA リソースグループ , 80
HP-UX, 78
Linux, 78
HP-UX
HP ServiceGuard, 78
NIS, 100
nnmharg.ovpl スクリプト , 119
仮想ホスト , 85, 92
共有ディスクの設定 , 97
共有ディスクフォーマット , 85, 93
コマンド , 114
ログファイル , 120
https
デフォルトポート , 76
プロトコル , 74
https 接続情報を NNMi ユーザーに通信 , 76
http プロトコル , 73

I

ICMP
- IPv4 アドレス
  - 無反応 , 249
  - 応答性 , 250
- State Poller, 38
- アドレスモニタリング , 63
- トラフィックの無効化 , 38
- プロトコル , 35
- 要求 , 43

ICMP への IPv4Address
- 応答性 , 250
- 無反応 , 249

InterfaceDown インシデント , 246

ipNoLookup.conf ファイル , 151

IPv4 アドレス , 244

IP アドレス
- HA 用に変更
  - NNMi, 99 ～ 102
  - NNM iSPI for Performance, 103
- NNMi 管理サーバーの変更 , 132
- 範囲 , 53 ～ 55

iSPI
- NNM iSPI for Performance
  - コンポーネント稼働状態 , 64

IWS、HPOM, 188

J

Java Client、AlarmPoint, 223

Java Runtime Environment、インストールする , 178

jboss
- ディレクトリ , 126
- バージョン , 74

jbossServer.log ファイル
- HP-UX, 120
- Linux, 121
- Solaris, 121
- UNIX, 116
- Windows, 116, 120

K

keytool コマンド , 74

L

Launcher、NNM 6.x/7.x, 183

license.txt ファイル , 100
- 管理サーバー , 132

LicFile.txt ファイル
- licenses.txt のアップデート , 98
- バックアップする , 86
  - NNMi データ , 127

Linux
- HPOM 統合 , 189, 214
- HP ServiceGuard, 78
- NIS, 100
- nnmharg.ovpl, 119
- 仮想ホスト , 85, 92
- 共有ディスクの設定 , 97
- 共有ディスクフォーマット , 85, 93
- コマンド , 114
- ログ ファイル , 120

lvm2 共有ディスクフォーマット , 85, 93

M

messages* ログ ファイル
- Solaris, 121

messages* ログファイル
- Linux, 120
- Solaris, 121

MIB II 変数 , 64

MIB ブラウザ
- NNM 6.x/7.x, 183
- URL 例 , 180

Microsoft Cluster Services, 78
- HA リソースグループ , 80
- nnmhamscs.vbs スクリプト , 119
- ダイナミック ディスク , 98

MINCAUSE アルゴリズム , 241

Mount_A.log ファイル , 121

MTTR, 226

N

netmon.cmstr ファイル , 148

NetScout Systems nGenius Performance Manager

netstat コマンド , 74

nGeniusNNM8.zip ファイル , 227

nGenius Performance Manager
- サポートされるバージョン , 226
- 統合
  - 使用法 , 227
  - トラブルシューティング , 226 ～ 228
  - 無効化 , 228
  - 有効化 , 227
- ドキュメント , 227

nGenius Performance Manager と NNMi の統合
使用法 , 227
トラブルシューティング , 226 ～ 228
無効化 , 228
有効化 , 227
NIS, 100
NmsApa サービス
Causal Engine, 241
オブジェクト ステータスの設定 , 242
設定変更 , 268
デバイスが発生したトラップ , 267
ネットワーク接続 , 267
ノード
更新済み , 267
到達不可能 , 263
nmsdbmgr.log ファイル
HP-UX, 120
Linux, 121
Solaris, 121
Windows, 120
nmsdbmgr サービス , 129
起動の問題 , 116
ディスク フェイルオーバー , 116
NNM 6.x/7.x
NNMi との統合
イベント転送 , 174 ～ 178
テスト , 180 ～ 183
トラブルシューティング , 183
リモートビューの起動 , 178 ～ 180
NNMi への移行
SNMP, 148 ～ 152
イベント , 167 ～ 172
オプション , 146 ～ 147
検出 , 152 ～ 161
ステータスモニタリング , 161 ～ 166
提案するパス , 145
フェーズ , 146
イベントノードリストの設定 , 176
イベントモニタリングのカスタマイズ , 142
管理ステーション , 181 ～ 182
ステータスモニタリング , 139
動的ビュー , 178
ネットワーク検出 , 136
ビュー , 178
NNM 6.x/7.x イベントビュー , 183
NNM 6.x/7.x からの移行
移行の割合を管理 , 173
イベント , 167 ～ 172
オプション , 146 ～ 147
検出 , 152 ～ 161
ステータスモニタリング , 161 ～ 166
提案するパス , 145
フェーズ , 146
NNM 6.x/7.x と NNMi の統合
イベント転送 , 174 ～ 178
テスト , 180 ～ 183
トラブルシューティング , 183
リモートビューの起動 , 178 ～ 180
NNM 6.x/7.x 動的ビューを起動 , 183
NNM 6.x/7.x を追加
管理ステーション , 176
ビュー , 179
nnm.envvars.bat コマンド , 236 ～ 238
nnm.envvars.sh コマンド , 236
nnmbackup.ovpl
コマンドラインオプション , 125
説明 , 124
プライマリクラスタノードの設定 , 86
nnmbackupembdb.ovpl
コマンドラインオプション , 129
説明 , 124
nnmcommconf.ovpl コマンド , 42
nnmconfigexport.ovpl
管理ステーション設定を保存 , 176
設定の XML への出力 , 132
nnmconfigimport.ovpl コマンド , 132
nnmdatareplicator.conf ファイル , 118
nnmdatareplicator.ovpl
コマンド , 97
スクリプト , 119
nnmdumpevents コマンド , 183
nnmhaclusterinfo.ovpl スクリプト , 118, 119
nnmhaconfigure.ovpl
コマンド , 116
スクリプト , 119
nnmhadisk.ovpl コマンド , 117
nmsdbmgr のトラブルシューティング , 116
設定ファイルのレプリケーション , 97
nnmhadisk.ovpl スクリプト , 119
nnmhamonitor.ovpl スクリプト , 119
nnmhamscs.vbs スクリプト , 119
nnmharg.ovpl スクリプト , 119
nnmhargconfigure.ovpl
コマンド , 114
スクリプト , 119
nnmhastart.ovpl スクリプト , 119
nnmhastartrg.ovpl スクリプト , 119
nnmhastop.ovpl スクリプト , 119
nnmhaunconfigure.ovpl スクリプト , 119

NNM iSPI for Performance
コンポーネント稼働状態 , 64
NNMi 管理サーバーの変更
IP アドレス , 132
概要 , 131
NNMi クイックスタート設定ウィザード , 15
NNMi コンソール
HPOM, 188
NNM 6.x/7.x インシデント設定を確認する , 177
アクセス , 188
セキュリティ , 73 ～ 76
トランザクションベースの更新 , 28
NNMi コンソール接続のセキュリティを確保, 73 ～ 76
NNMi 設定移動の準備 , 131
NNMi と HPOM の統合
値 , 188
概要 , 187
サポートされるバージョン , 189
使用法 , 193
トラブルシューティング , 195 ～ 197
ドキュメント、HPOM, 189
無効化 , 194
有効化
UNIX, 191
Windows, 190
NNMi に AlarmPoint を統合
概要 , 221 ～ 223
使用法 , 223
トラブルシューティング , 224
無効化 , 224
有効化 , 223
nnmlicense.ovpl コマンド , 98 ～ 100, 132
管理サーバーの変更 , 132
nnmloadseeds.ovpl コマンド , 53
nnmrestore.ovpl
コマンドラインオプション , 128
説明 , 124
nnmrestoreembdb.ovpl
コマンドラインオプション , 129
説明 , 124
NOC, 149
NodeDown インシデント , 246
NodeOrConnectionDown インシデント , 246
Nortel
スイッチ
階層 , 31
ノードグループの定義 , 30
ルーター
階層 , 31
ノードグループの定義 , 30
nslookup コマンド , 84, 92

O

oid_to_sym file, 152
OLDsyslog.log ファイル , 120
OM
OpenView アプリケーションサーバー、
NNM 6.x/7.x, 179
Operations Manager
Oracle データベースと HA, 95
ov.conf ファイル , 116, 118
OV_Message イベント、NNM 6.x/7.x, 182
ovaddr コマンド、NNM 6.x/7.x, 179
ovalarm の URL 例 , 180
ovas, 179
ovas、NNM 6.x/7.x, 179
ovdumpevents コマンド、NNM 6.x/7.x, 183
ovet*.log ファイル
HP-UX, 120
Linux, 121
Solaris, 121
Windows, 120
OV jovw の URL 例 , 180
OV Launcher の URL 例 , 180
ovspmd.log ファイル
HP-UX, 120
Linux, 120
Solaris, 121
Windows, 120
ovspmd サービス
トラブルシューティング , 115
ovstart コマンド
トラブルシューティング , 115
ovstop コマンド
トラブルシューティング , 115
フェイルオーバーを行わせないように , 102, 111
ovtopodump コマンド、NNM 6.x/7.x, 181
ovtopofix コマンド、NNM 6.x/7.x, 181
ovwId、ノード詳細、使用している , 180
ovw、NNM 6.x/7.x, 183
OV アプリケーションサーバー、NNM 6.x/7.x, 179

P

PC、管理対象としない , 45
Performance Insight, 23

ping, 49
コマンド , 247 ～ 254
プロトコル , 35
要求 , 64
pmd の起動の問題 , 116
postgres.log ファイル
HP-UX, 120
Linux, 120
Solaris, 121
Windows, 120

R

remsh コマンド , 78
Report Presenter の URL 例 , 180

S

-scope all, 127
-scope config, 126
-scope event, 127
-scope topology, 127
sendMsg.ovpl コマンド、NNM 6.x/7.x, 181 ～ 182
server.xml ファイル , 74 ～ 76
SLA, 63
SNMP
Data Presenter, 180
MIB ディレクトリ , 127
NNMi への移行 , 148 ～ 152
アクセスの設定 , 148 ～ 152
エージェントクエリー
応答性 , 248
無反応 , 247
エージェントステータス , 244
コミュニティ文字列 , 36
コンポーネント稼働状態 , 64
設定の調整 , 43
通信の問題 , 54
デバイスアクセス , 42
ノードの設定 , 42
バージョンの優先 , 36
プロトコル , 35, 243
無効化
通信 , 40
トラフィック , 38
モニタリング , 64
要求 , 43
snmpout.txt ファイル , 149
snmptrap コマンド、NNM 6.x/7.x, 182
SNMP 情報の移行 , 148 ～ 152
Solaris
NIS+, 100
nnmharg.ovpl スクリプト , 119
nnmhargconfigure.ovpl
コマンド , 114
スクリプト , 119
Veritas Cluster Server, 78
仮想ホスト , 85, 92
共有ディスクの設定 , 97
共有ディスクフォーマット , 85, 93
ssh コマンド , 78
SSL テクノロジ , 73
State Poller
ICMP, 38
確認
稼働状態統計 , 68
通信設定 , 43
概念 , 57 ～ 58
計画作成 , 58 ～ 64
症状 , 242
設定 , 64 ～ 66
設定の評価 , 66 ～ 68
調整 , 68 ～ 69
syslog.log ファイル , 120

T

tar ファイル , 126, 128
Tomcat 設定 , 73
traceroute コマンド , 40
trapd.conf の編集 , 176
trapd.conf ファイル、編集する , 176
trapd.conf を手動で編集する , 176

U

UNIX
- HA 設定
  - スクリプト , 118
  - ファイル , 118
  - ログファイル , 120
- HPOM 統合
  - バージョン , 189
  - 有効化 , 191
- ipNoLookup.conf コマンド , 151
- jboss コマンド , 74
- LicFile.txt ファイル
  - HA クラスタライセンスのインストール , 98
  - プライマリ HA クラスタノードの設定 , 86
- netmon.cmstr コマンド , 148
- nGeniusNNM8.zip ファイル , 227
- nnm8EventForwardDestinations.txt
  - ファイル , 175
- nnm.envvars.sh コマンド , 236
- nnmlicense.ovpl コマンド , 98
- nnmtrapd ログファイル , 116
- pmd ログファイル , 116
- remsh コマンド , 78
- sendMsg.ovpl コマンド , 182
- ssh コマンド , 78
- trapd.conf ファイル , 176
- 環境変数
  - インストール , 16
  - 管理 , 237
- 共有ディスクのディレクトリ , 97
- ディスクグループ , 85, 93
- ボリュームグループ , 85, 93
- ポート番号 , 179
- マウントポイント , 85, 93

URL
- 選択が必要 , 180
- 選択を必要としない , 180

UUID、ノード詳細、使用している , 180

V

Veritas Cluster Server, 78
- HA リソースグループ , 80
- nnmharg.ovpl スクリプト , 119

Volume_A.log ファイル , 121

vxfs 共有ディスクフォーマット , 85, 93

W

Web サーバー
- AlarmPoint, 223
- https プロトコルの有効化 , 73
- パブリックキー証明書 , 73
- ポート , 179

Windows
- HA 設定
  - スクリプト , 118
  - ファイル , 118
  - ログファイル , 120
- HPOM 統合
  - バージョン , 189
  - 有効化 , 190
- ipNoLookup.conf コマンド , 151
- jboss コマンド , 74
- LicFile.txt ファイル
  - HA クラスタライセンスのインストール , 98
  - プライマリ HA クラスタノードの設定 , 86
- Microsoft Cluster Services, 78
- netmon.cmstr コマンド , 148
- nGeniusNNM8.zip ファイル , 227
- nnm8EventForwardDestinations.txt
  - ファイル , 175
- nnm.envvars.bat コマンド , 236 ～ 238
- nnmlicense.ovpl コマンド , 98
- nnmtrapd ログファイル , 116
- pmd ログファイル , 116
- sendMsg.ovpl コマンド , 182
- trapd.conf ファイル , 176
- 仮想ホスト , 85, 92
- 環境変数
  - インストール , 16
  - 管理 , 236 ～ 238
- 共有ディスクの設定 , 97
- 共有ディスクのディレクトリ , 96
- 共有ディスクフォーマット , 85, 93
- コマンド
  - nnmhargconfigure.ovpl, 114
- ポート番号 , 179
- マウントポイント , 85, 93

X

XML ファイル , 34, 132

xnmevents コマンド、NNM 6.x/7.x, 175

XServer, 176

あ

アーカイブファイル , 126

アーキテクチャ、HA クラスタ , 79

アクションメニュー , 183

アクセス資格認定
- コミュニティ文字列 , 36
- 設定 , 37
- 複数 , 41

アクティブ
- プロトコル , 40

アドレス、NNMi 管理サーバー
変更 , 132
アドレス、管理サーバー
確認 , 43
優先 , 38
アプリケーション
アラート生成 , 221
サーバー、AlarmPoint, 223
アプリケーションアラートアラームカテゴリ、NNM 6.x/7.x, 178
アプリケーションステータスインシデント
カテゴリ , 178
アラート生成アプリケーション , 221
アラーム、NNM 6.x/7.x
カテゴリ , 177
動的ビューを起動 , 183
マッピング , 177 ～ 178
安全なソケットレイヤー テクノロジ , 73

い

移動
NNMi
管理サーバー , 131
設定 , 132
インタフェース , 267
イベント
NNM 6.x/7.x からの移行 , 167 ～ 172
OV_Message、NNM 6.x/7.x, 182
削減 , 176
転送
確認する , 181
管理サーバー , 174
トラップとしてのシステム , 242
ノード / インタフェース停止 / 稼働を発生 , 181
モニタリング
カスタマイズ , 142
概念 , 144
領域データ , 127
イベント設定ウィンドウ、NNM 6.x/7.x, 174
インシデント
NNM 6.x/7.x を NNMi コンソールで確認する , 177
カテゴリ , 177 ～ 178
定義 , 246
転送 , 188
フィルタ , 200 ～ 202
ブラウザ , 173
インシデント Web サービス、HPOM, 188
インシデント設定フォーム、設定を確認する , 177
インシデントワークスペース , 181
インストール
AlarmPoint Java Client, 223
HA クラスタ用のライセンス , 98
HP Performance Insight, 23
Java Runtime Environment, 178
パブリックキー証明書 , 73 ～ 74
インタフェース
HA 設定の仮想ホストネットワーク , 85, 92
移動 , 267
稼働
管理 , 254
操作 , 252
グループ
確認 , 66
事前設定 , 62
設定 , 64
フィルタリング , 33
目的 , 29
ステータス , 244
設定 , 33
停止
管理 , 253
操作 , 251
モデル , 28
モニタリング , 59
モニタリングの設定 , 65
［インタフェースグループ］フォーム , 61
［インタフェースグループ］ワークスペース , 61
［インタフェース設定］フォーム , 69

う

ウィザード、NNMi クイックスタート設定 , 15
ウィンドウ、［HP Network Node Manager i-series］について , 68

え

エージェント、SNMP
応答性 , 248
ステータス , 244
無反応 , 247
エピソード , 243
エラーアラームカテゴリ、NNM 6.x/7.x, 178
エンドツーエンドの診断 , 241

お

応答性
ICMP への IPv4Address, 250
クエリーする SNMP エージェント , 248
オフラインバックアップ , 125 ～ 126
オブジェクトグループ、定義の作成 , 62

オブジェクト、ステータス設定 , 242
オプション、NNM 6.x/7.x からの移行 , 146 ～ 147
オンラインバックアップ , 125 ～ 126

か

解決、名前の制限 , 151
階層、ノードグループ , 31
拡張モニタリング , 59
確認
SNMP 用に設定されたノード , 42
State Poller
稼働状態統計 , 68
通信設定 , 43
インタフェースグループ , 66
管理 IP アドレス , 43
順序番号 , 65
通信設定 , 42
デバイスについての SNMP アクセス , 42
デフォルト設定 , 66
ネットワーク
モニタリングの設定 , 66
ノードグループ , 66
モニタリングの設定 , 66 ～ 67
確認する
NNM 6.x/7.x インシデント設定 , 177
イベント転送 , 181
ネットワーク
接続 , 54
カスタマイズ
イベントモニタリング , 142
カスタムインシデント属性 , 179
仮想 IP アドレスのサポート , 78
仮想ホスト、HA 設定
ネットマスク , 84, 92
ネットワークインタフェース , 85, 92
短い名前 , 84, 92
カテゴリ
ステータス , 244
マッピング , 177
カテゴリのマッピング , 177
稼働
インタフェース
管理 , 254
操作 , 252
接続
操作 , 256
ルーター , 264
ノード
シャドウの除去 , 262
分散ルーター , 258
ルーター , 264
環境、テスト , 15
環境変数
UNIX
インストール , 16
管理 , 237
Windows
インストール , 16
管理 , 236 ～ 238
概要 , 235
管理
アドレス
確認 , 43
優先 , 38
サーバー
AlarmPoint Java Client のインストール , 223
NNM 6.x/7.x イベントの転送 , 174
NNMi の移動 , 131
サーバー、HPOM 統合の無効化 , 194
ステーション
NNM 6.x/7.x エンティティを作成 , 178
NNM 6.x/7.x をトポロジに追加 , 176
設定を保存 , 176
設定変更 , 268
管理ステーション設定を保存 , 176
管理ステーションフォーム , 179
管理対象ノード、解析 , 243
管理対象ノードの解析 , 243
関連ドキュメント , 18
概念
Causal Engine, 242
HA, 79
イベントモニタリング , 144
検出 , 138
状態ポーリング , 57 ～ 58
ステータスモニタリング , 141
設定 , 27
通信 , 35
外部リレーショナルデータベース , 123

き

キー、パブリック , 73 ～ 74

基幹、ネットワーク , 45
ルール、自動検出
順序 , 48
評価 , 55 ～ 56
ルールベース検出
概要 , 47
評価 , 55 ～ 56
有効化 , 55
起動
HA メンテナンス後の NNM iSPI for Performance, 103
HA メンテナンス後の NNMi, 102
HA リソースグループ , 114
起動の問題
nmsdbmgr, 116
NNMi, 115
pmd, 116
基本共有ディスクフォーマット , 85, 93
キャッシュ、ARP, 51
共存、HP Performance Insight, 23
共有 HA データ , 96 ～ 97
共有される設定データ , 126
共有ディスク
サポート , 78
設定 , 97
データ , 96
データファイルのコピー , 84
ファイル , 117
マウント解除 , 106, 108
共有ディスクのマウント解除 , 106, 108
共有ファイルシステムのタイプ、HA 設定 , 85, 92

く

クイックスタート設定ウィザード , 15
組み込みデータベース
バックアップおよびリストアする , 128
リレーショナル , 123
クラスタノード、HA の設定
NNMi
セカンダリ , 88
プライマリ , 86 ～ 88
NNM iSPI for Performance
セカンダリ , 94
プライマリ , 93 ～ 94
グループ
インタフェース
フィルタリング , 33
目的 , 29
ディスク , 85, 93
ノード , 29
ボリューム , 85, 93

け

計画作成
状態ポーリング , 58 ～ 64
通信 , 39
ポーリング間隔 , 63
検出
NNM 6.x/7.x
移行 , 152 ～ 161
比較 , 136
再スタート , 33
重要概念 , 138
スイッチ , 55
スパイラル , 45
デメリット , 48
トランザクションベースの更新、例外 , 28
ノードを削除 , 54
パフォーマンス , 43
評価
ルールベース検出 , 55 ～ 56
リストベース検出 , 54
方法
ルールベース検出 , 47 ～ 48
リストベース検出 , 47
ルーター , 55
原因、こんぽん , 242

こ

高可用性 －*参照* HA クラスタ ; HA リソースグループ
恒久ライセンスのパスワード , 100
更新
server.xml, 74 ～ 76
更新中
ノード , 267
構文、バックアップスクリプト , 125
コネクタブロック、https 設定 , 76

コマンド
cluster.exe, 97
hostname, 100
keytool, 74
netstat, 74
nnm.envvars.bat, 236 ～ 238
nnm.envvars.sh, 236
nnmcommconf.ovpl, 42
nnmconfigexport.ovpl
管理ステーション設定を保存 , 176
設定の XML への出力 , 132
nnmconfigimport.ovpl, 132
nnmdatareplicator.ovpl, 97
nnmdumpevents, 183
nnmhaconfigure.ovpl, 116
nnmhadisk.ovpl
nmsdbmgr のトラブルシューティング , 116
設定ファイルのレプリケーション , 97
nnmhargconfigure.ovpl, 114
nnmlicense.ovpl, 98 ～ 100
管理サーバーの変更 , 132
nnmloadseeds.ovpl, 53
nslookup, 84, 92
ovaddr, 179
ovspmd
トラブルシューティング , 115
ovstart
トラブルシューティング , 115
ovstop
トラブルシューティング , 115
フェイルオーバーを行わせない
ように , 102, 111
ovtopodump, 181
ovtopofix, 181
ping, 247 ～ 254
remsh, 78
sendMsg.ovpl, 181 ～ 182
snmptrap, 182
ssh, 78
traceroute, 40
xnmevents, 175
コマンドラインオプション
nnmbackup.ovpl, 125
nnmbackupembdb.ovpl, 129
nnmrestore.ovpl, 128
nnmrestoreembdb.ovpl, 129
コミュニティ文字列
アクセス資格認定 , 36
複数 , 41
コンソール。
コンテインメント、ノードグループ , 31
コンポーネント稼働状態モニタリング , 64 ～ 69
コンポーネント登録ファイル , 126
根本原因 , 242
結論を生み出す , 242

さ

サーバー
AlarmPoint, 223
サービス
NmsApa
ステータス , 242
設定変更 , 268
デバイスが発生したトラップ , 267
ネットワーク接続 , 267
ノードは到達不可能 , 263
ノードを更新 , 267
サービスレベル契約条項 , 63
再起動
HA メンテナンス後の NNM iSPI for
Performance, 103
再試行
値 , 40
設定 , 37
調整 , 43
再スタート
HA メンテナンス後の NNMi, 102
検出 , 33
削減
イベント , 176
デフォルトコミュニティ文字列 , 43
認証不合格 , 43
削除
検出されたノード , 54
ライセンスキー , 132
作成者属性 , 28
作成中
NNM 6.x/7.x 管理ステーションエンティティ , 178
オブジェクトグループ定義 , 62
再使用可能なノードグループ , 63
シャドウ , 261
サポートされるバージョン
AlarmPoint 統合 , 222
HPOM 統合 , 189
HP UCMDB 統合 , 214
nGenius Performance Manager 統合 , 226

し

シード、ルールベース検出 , 47
資格認定、アクセス
コミュニティ文字列 , 36
設定 , 37
複数 , 41
しきい値アラームカテゴリ、NNM 6.x/7.x, 178

システム
オブジェクト ID 範囲
自動検出 , 53
評価 , 56
共有ファイルタイプ、HA 設定 , 85, 92
デバイス対応マトリックス , 21
リリース , 69
失敗
ネットワークシナリオ , 246 ～ 264
シナリオ
HA クラスタ , 80
ネットワーク失敗 , 246 ～ 264
シャドウ
作成中 , 261
除去 , 262
シャドウの除去 , 262
詳細、ノード , 180
証明書、パブリックキー , 73 ～ 74
自己署名証明書
server.xml の更新 , 75
パブリックキー証明書のインストール , 74
事前設定
インタフェースグループ , 62
ノードグループ , 63
自動検出ルールの順序 , 48
順序属性
自動検出ルール , 48
ベストプラクティス , 28
順序番号、確認 , 65
順序、評価 , 58

す

スィープ , 49
スイッチ
階層 , 31
検出 , 55
デフォルト , 56
ノードグループの定義 , 30
スクリプト
HA 設定 , 118
nnmhaclusterinfo.ovpl, 118
データのリストア , 127
バックアップ , 125 ～ 127
ステータス
SNMP エージェント , 244
インシデントカテゴリ , 178
インタフェース , 244
オブジェクト , 242
ノード , 244
ノードグループ , 245
ポーリング
開始 , 67
パフォーマンスの評価 , 67 ～ 68
モニタリング
NNM 6.x/7.x からの移行 , 161 ～ 166
概念 , 141
設定 , 139
ステータスアラームカテゴリ、NNM 6.x/7.x, 178
ステータスインシデントカテゴリ , 178
ステータスポーリングの開始 , 67
スパイラル検出 , 45

せ

制限のある環境 , 15
製品、HA, 78
セカンダリ HA クラスタノード
設定
NNMi, 88
NNM iSPI for Performance, 94
設定情報 , 84

設定
HA クラスタ
NNMi, 84 ～ 88
NNM iSPI for Performance, 91 ～ 95
HA クラスタノード、セカンダリ
NNMi, 88
NNM iSPI for Performance, 94
HA クラスタノード、プライマリ
NNMi, 86 ～ 88
NNM iSPI for Performance, 93 ～ 94
HA 情報
NNMi, 84
NNM iSPI for Performance, 91
HA のトラブルシューティング , 113 ～ 117
HPOM 統合 , 198 ～ 203
NNM 6.x/7.x
イベントの転送 , 174
ビューを追加 , 179
NNMi 移動の準備 , 131
NNMi の移動 , 132
SNMP アクセス , 148 ～ 152
Tomcat, 73
アクセス資格認定 , 37
インシデントカテゴリ , 178
インタフェース
グループ , 64
モニタリング , 65
オブジェクトステータス , 242
概念 , 27
ルールベース検出 , 53
共有ディスク , 97
共有データ , 126
再試行 , 37
状態ポーリング , 64 ～ 66
スクリプト、HA クラスタ , 118
設定 , 33
タイムアウト , 37
通信 , 41, 42
通信に意味のあるデフォルト , 39
デフォルトの確認 , 66
トランザクションベースの更新 , 28
ネットワークモニタリングの確認 , 66 ～ 67
ノード , 40
グループ , 64
モニタリング , 65
比較
イベントモニタリングのカスタマイズ , 142
ステータスモニタリング , 139
ネットワーク検出 , 136
ファイル
HA クラスタ , 118
HA 設定 , 118
HA 用のレプリケーション , 97
変更
管理 , 268
ネットワーク , 267
保存
管理ステーション , 176
既存 , 28
ポーリングの例 , 59
モニタリング , 67
モニタリングのバッチ処理 , 64
やり直し , 33
リストベース検出 , 53
領域 , 39
領域データ , 126
レベル , 37
ワークスペース , 179
設定アラームカテゴリ、NNM 6.x/7.x, 178
設定のやり直し , 33
［設定］ワークスペース
状態ポーリングの設定 , 64
状態ポーリングの評価 , 66
接続
https 情報の通信 , 76
稼働
操作 , 256
ルーター , 264
停止
操作 , 255
ルーター , 264
接続、ネットワークを確認する , 54
前提条件
HA クラスタ用の NNMi, 78
ソフトウェア , 21
ハードウェア , 21

そ

そのまま使用できるポーリング動作 , 59
ソフトウェア
対応 , 21
要件 , 19
属性
作成者 , 28
順序 , 28

た

対応
HA 製品 , 78
ハードウェアとソフトウェア , 21
タイムアウト , 36
値 , 40
設定 , 37
調整 , 43
タスクフローモデル , 27

「多対多」の配置 , 187
第三者証明書 , 74
ダイナミック ディスク , 98

ち

チェックリスト、ポーリング , 58
調整
　状態ポーリング , 68 ～ 69
　通信 , 43

つ

追加のノードパックライセンスを購入する , 54
通信
　概念 , 35
　計画作成 , 39
　設定 , 41, 42
　設定の確認 , 43
　設定の評価 , 41
　設定領域 , 39
　調整 , 43
通信設定に意味のあるデフォルト , 39

て

停止
　HA フェイルオーバーを行わせないように NNM iSPI for Performance を , 103
　HA フェイルオーバーを行わせないように NNMi を , 102
　HA リソースグループ
　　NNMi, 105
　　NNM iSPI for Performance, 108
　インタフェース
　　管理 , 253
　　操作 , 251
　接続 , 255
　ノード
　　シャドウの作成 , 261
　　分散ルーター , 257
　　ルーター , 264
テーブル、設定 , 126
テスト
　NNM 6.x/7.x
　　統合 , 180
　　トラップ , 182
　通信設定 , 41
　ラグ タイム , 40
テスト環境 , 15
転送、イベント
　NNM 6.x/7.x から NNMi 管理サーバーへ , 174
　確認する , 181
転送先リストファイル , 175
転送されたインシデント、HPOM のトラブルシューティング , 195 ～ 197
ディスク
　共有
　　ディレクトリ , 96
　　データファイルのコピー , 84
　グループ、HA 設定 , 85, 93
　フェイルオーバー , 116
ディレクトリ
データ
　イベント領域 , 127
　共有、設定 , 126
　共有ディスク , 96
　収集、State Poller, 64
　収集の確認 , 67
　設定領域 , 126
　トポロジ領域 , 127
　バックアップする
　　完全バックアップ , 127
　　組み込みデータベースのみ , 128
　　スクリプト , 125 ～ 127
　　方針 , 123 ～ 124
　リストアする
　　組み込みデータベースのみ , 128
　　スクリプト , 127
データの収集
　ステータスポーリング
　　確認 , 67
　　データの選択 , 64
データのリストア
　組み込みデータベース , 128
　スクリプト , 127
データベース
　組み込みをバックアップする , 128
　組み込みをリストアする , 128
　トポロジ , 57
　リセット , 33
デバイス
　SNMP アクセス , 42
　発生したトラップ , 267
　フィルタ , 32
　プロファイル
　　概念 , 32
　　状態ポーリング , 63
デバイスプロファイルなし , 152
デバイスを検出から除外 , 48

デフォルト
https ポート , 76
意味のある、通信 , 39
ルールベース検出 , 53
検出 , 45
コミュニティ文字列 , 43
スイッチ , 56
設定 , 33
設定の確認 , 66
ルーター , 56
デメリット、検出 , 47

と

統合モジュール、HPOM, 188
到達可能なノード , 263
到達不可能なノード , 263
トポロジ
NNM 6.x/7.x 管理ステーション , 176
概要 URL 例 , 180
データベース , 57
領域データ , 127
トラップ
NNM 6.x/7.x システムをテスト , 182
発生中 , 267
トラフィック
制御 , 40
無効化 , 38
トラフィックの制御 , 40
トラブルシューティング
AlarmPoint 統合 , 224
HA 設定 , 113 ～ 117
HPOM 統合 , 195 ～ 197
同期 , 197
ファイアウォール , 197
nGenius Performance Manager 統合 , 226 ～ 228
検出
自動検出 , 55 ～ 56
同期、HPOM, 197
同期、HPOM のトラブルシューティング , 197
動作、そのまま使用できるポーリング , 59
動的ビュー
NNM 6.x/7.x, 178
NNMi から起動 , 183
ドキュメント
AlarmPoint, 223
nGenius Performance Manager, 227
関連 , 18
ハードウェアとソフトウェアの要件 , 21
場所 , 145
ドメインネームサービス , 100

な

名前解決、制限 , 151
名前解決の制限 , 151

に

認証不合格、削減 , 43

ね

ネットマスク、HA 設定の仮想ホスト , 84, 92
ネットワーク
アクセスの無効化 , 38
確認
設定 , 66
確認する
接続 , 54
基幹 , 45
検出
NNM 6.x/7.x, 136
評価 , 54 ～ 56
失敗のシナリオ , 246 ～ 264
負荷 , 69
モニタリング
設定 , 66
動作 , 59
ノード , 69
ネットワークインタフェース、HA 設定の仮想ホスト , 85, 92
ネットワーク運営センター , 149
ネットワーク情報サービス , 100
ネットワーク設定の変更 , 267
ネットワーク待ち時間 , 36

の

ノード
- HA クラスタのセカンダリの設定
  - NNMi, 88
  - NNM iSPI for Performance, 94
- HA クラスタのプライマリの設定
  - NNMi, 86 ～ 88
  - NNM iSPI for Performance, 93 ～ 94
- SNMP 設定 , 42
- 稼働
  - router, 264
  - シャドウの除去 , 262
  - 分散ルーター , 258
- 管理対象の解析 , 243
- 更新中 , 267
- 削除する、検出した , 54
- 詳細 , 180
- ステータス , 244
- 設定 , 33, 40
- 停止
  - シャドウの作成 , 261
  - 接続 , 264
  - 分散ルーター , 257
- 到達可能 , 263
- 到達不可能 , 263
- モニタリング , 69
- モニタリングの設定 , 65
- レベルのフィルタリング , 176

ノード / インタフェース停止 / 稼働イベント , 181

ノードインベントリビュー , 176

ノードグループ
- インタフェースグループ , 33
- 階層 , 31
- 確認 , 66
- 事前設定 , 63
- ステータス , 245
- 設定 , 64
- 定義 , 30
- デバイスフィルタ , 32
- 非 SNMP デバイス , 60
- フィルタリング , 29
- メンバーシップ , 31

ノードグループの定義 , 30

［ノードグループ］フォーム , 61

［ノードグループ］ワークスペース , 61

［ノードの設定］フォーム , 69

は

ハードウェア
- 対応 , 21
- 必要な , 19

発生中
- トラップ , 267
- ノード / インタフェース停止 / 稼働イベント , 181

範囲、IP アドレス , 55

バージョン
- SNMP 優先 , 36
- 情報ファイル , 126

バージョン、HPOM 統合 , 189

バージョン、サポートされる
- AlarmPoint, 222
- HPOM, 189
- HP UCMDB, 214
- nGenius Performance Manager, 226

バージョン比較
- カスタマイズ
  - イベントモニタリング , 142
- ステータスモニタリング , 139
- ネットワークの検出 , 136

場所
- jboss 配備ディレクトリ , 126
- ipNoLookup.conf, 151
- LicFile.txt ファイル , 127
- netmon.cmstr, 148
- nGeniusNNM8.zip, 227
- SNMP MIB, 127
- trapd.conf, 176
- 環境変数
  - インストール , 16
  - 管理 , 236 ～ 238
- ドキュメント , 145
  - AlarmPoint, 223
  - nGenius Performance Manager, 227

バックアップ
- スクリプト
  - データのリストア , 127
  - データをバックアップする , 125
- ファイル , 129

バックアップする
- データ
  - 完全バックアップ , 127
  - 組み込みデータベースのみ , 128
  - スクリプト , 125 ～ 127
  - 方針 , 123 ～ 124

パス、NNM 6.x/7.x からの移行 , 145

パスワード、恒久ライセンス , 100

パソコン、管理対象としない , 45

パフォーマンスインシデントカテゴリ , 178

パフォーマンス、ステータスポーリング , 67 ～ 68

パブリックキー証明書、インストール , 73 ～ 74

パブリックキー証明書の取得 , 73 ～ 74

パラメータ
HPOM 統合設定 , 198 ～ 202
HP UCMDB 統合設定 , 217 ～ 219

ひ

非 SNMP デバイスノードグループ , 60
評価
検出の設定 , 54
状態ポーリング設定 , 66 ～ 68
通信設定 , 41
評価の順序 , 58
ヒント
IP アドレス範囲 , 55
システム ID 範囲 , 56
ヒント、設定
シード検出 , 53
自動検出 , 53
ビュー
NNMi から動的を起動 , 183
追加の NNM 6.x/7.x を設定 , 179

ふ

ファイアウォール
トラフィックの制御 , 40
ネットワークアクセスの無効化 , 38
ファイアウォール、HPOM 統合のトラブルシューティング , 197
ファイル
<resource_group>.cntl.log, 120
HA 設定
スクリプト , 118
ファイル , 118
ログファイル , 120
ipNoLookup.conf, 151
jbossServer.log, 116
license.txt, 100
管理サーバー , 132
LicFile.txt, 86, 98
バックアップ , 127
netmon.cmstr, 148
nGeniusNNM8.zip, 227
nmsdbmgr.log, 120
nnmdatareplicator.conf, 118
oid_to_sym, 152
OLDsyslog.log, 120
ov.conf, 116
HA 設定 , 118
ovspmd.log, 120
postgres.log, 120
server.xml, 74 ～ 76
snmpout.txt, 149
syslog.log, 120
tar, 128
trapd.conf, 176
XML, 132
アーカイブ , 126
共有ディスク , 117
クラスタノードで更新されない , 113
コンポーネント登録 , 126
システムタイプ , 85, 92
転送先リスト , 175
バージョン情報 , 126
バックアップ , 129
バックアップする , 123
レプリケーション , 97
ファイルシステムのタイプ、HA 設定 , 85, 93
フィルタ
インシデント , 200 ～ 202
デバイス , 32
ノードレベルの設定 , 176
フィルタリング
インタフェースグループ , 33
ノードグループ , 29
フェイルオーバー、ディスク , 116
フェーズ、NNM 6.x/7.x からの移行 , 146

フォーム
インシデントの設定 , 177
インタフェースグループ , 61
［インタフェース設定］, 69
管理ステーション , 179
ノードグループ , 61
［ノードの設定］, 69
モニタリングの設定 , 57, 69
不合格
認証の削減 , 43
フローモデル、タスク , 27
部分バックアップ , 125
ブラウザ、インシデント , 173
プライマリ HA クラスタノード
設定
NNMi, 86 ～ 88
NNM iSPI for Performance, 93 ～ 94
設定情報 , 84, 91
プリンタ、管理対象としない , 45
プロトコル
http, 73
SNMP, 243
アクティブ , 40
通信 , 35
ポーリング , 38
プロファイル、デバイス
概念 , 32
状態ポーリング , 63

へ

変更、設定
管理 , 268
ネットワーク , 267
変数、MIB II, 64
変数、かんきょう
概要 , 235
変数、環境
UNIX
MANPATH, 22
インストール , 16
管理 , 237
Windows
NNMi 8.00, 238
インストール , 16
管理 , 236
ベストプラクティス
NNMi 設定移動の準備 , 131
概要 , 15
既存の設定を保存 , 28
作成者属性 , 28
作成中
オブジェクトグループ定義 , 62
再使用可能なノードグループ , 63
順序属性 , 28
順序番号の確認 , 65
順序番号のダブルチェック , 41
デバイスプロファイルなし , 152
短いポーリング間隔 , 63
モニタリング設定のバッチ処理 , 64
優先される管理アドレス , 38
ページ、HA テキストファイル (Windows), 83
ページ、HA マンページ (UNIX), 83

ほ

方針、バックアップ , 123 ～ 124
ホームベース、NNM 6.x/7.x, 183
ホスト、HA 設定用の仮想
NNMi, 84
NNM iSPI for Performance, 92
ホスト名、HA 用に変更
NNMi, 99 ～ 102
NNM iSPI for Performance, 103
ボリュームグループ、HA 設定 , 85, 93
ポイント、マウント , 85, 93
ポート番号、Web サーバー , 179
ポーリング
間隔の計画作成 , 63
ステータス
開始 , 67
調整 , 68 ～ 69
パフォーマンスの評価 , 67 ～ 68
設定の例 , 59
そのまま使用できる動作 , 59
チェックリスト , 58
プロトコル , 38

ま

毎週のデータバックアップ , 123
マウントポイント、HA 設定 , 85, 93
待ち時間、ねっとわーく , 36
末端ノード、管理対象としない , 45
マンページ、HA, 83

み

短い名前、HA 設定の仮想ホスト , 84, 92

む

無効化
AlarmPoint 統合 , 224
nGenius Performance Manager 統合 , 228
SNMP, 40
トラフィック , 38
ネットワークアクセス , 38
無反応
ICMP への IPv4Address, 249
クエリーする SNMP エージェント , 247

め

メモリーリソース , 69
メリット
ルールベース検出 , 47
リストベース検出 , 47
メンバーシップ、ノードグループ , 31

も

文字列、コミュニティ
アクセス資格認定 , 36
複数 , 41
モデル
タスクフロー , 27
ユーザーインタフェース , 28
モニタリング
イベント
カスタマイズ , 142
概念 , 144
インタフェース , 65
拡張 , 59
ステータス
概念 , 141
設定 , 139
設定 , 67
確認 , 66 ～ 67
設定 , 33
バッチ処理 , 64
ネットワーク
設定の確認 , 66
ノード , 69
ノードの設定 , 65
モニタリング設定のバッチ処理 , 64
［モニタリングの設定］フォーム
ステータスポーリングの調整 , 69
説明 , 57
ポーリングの種類と間隔の設定 , 61
問題、HA の起動
nmsdbmgr, 116
NNMi, 115
pmd, 116

ゆ

有効化
AlarmPoint 統合 , 223
HPOM 統合
UNIX, 191
Windows, 190
https プロトコル , 73
nGenius Performance Manager 統合 , 227
ルールベース検出 , 55
ユーザー、https 接続情報を通信 , 76
ユーザーインタフェースモデル , 28
優先
SNMP のバージョン , 36
管理アドレス , 38

よ

要件、ハードウェアとソフトウェア , 21
用語
HA, 80

ら

ライセンス
HA クラスタ , 98
限度 , 48
購入する、追加のノードパック , 54
ディレクトリの場所 , 127
内容 , 46
ライセンスキーの削除 , 132
ラグタイム、テスト , 40

り

リストアスクリプト , 127
リストベース検出 , 54
概要 , 47
評価 , 54
リセット
設定 , 33
リソースグループ、HA, 84, 92
リリース、システム , 69
利点
ルールベース検出 , 47
リストベース検出 , 47
リファレンスページ、HA, 83

領域
イベント , 127
設定 , 126
トポロジ , 127
領域、設定 , 39
リリースノート , 21
リレーショナルデータベース、バックアップする , 123

る

ルーター
階層 , 31
検出 , 55
デフォルト , 56
ノードグループの定義 , 30
モニタリング , 59

れ

例
HPOM 統合インシデント フィルタ , 202
SNMP 情報 , 148
ノードグループの設定 , 65
ポーリング設定 , 59
レプリケーション、設定ファイル , 97
レベル、設定 , 37

ろ

ログファイル、HA クラスタ
更新されない , 113
設定 , 120

わ

ワークスペース
インシデント , 181
インタフェースグループ , 61
設定
NNM 6.x/7.x 管理ステーションを設定 , 179
状態ポーリングの設定 , 64
状態ポーリングの評価 , 66
ノードグループ , 61