

HP OpenView Performance Insight

Interface Reporting ifEntry Datapipe ユーザーガイド

ソフトウェアバージョン : 2.1

Reporting and Network Solutions 7.0



2005 年 6 月

© Copyright 2004 Hewlett-Packard Development Company, L.P.

ご注意

1. 本書に記載した内容は、予告なしに変更することがあります。
2. 当社は、本書に関して特定目的の市場性と適合性に対する保証を含む一切の保証をいたしかねます。
3. 当社は、本書の記載事項の誤り、またはマテリアルの提供、性能、使用により発生した直接損害、間接損害、特別損害、付随的損害または結果損害については責任を負いかねますのでご了承ください。
4. 本製品パッケージとして提供した本書、**CD-ROM**などの媒体は本製品用だけにお使いください。プログラムをコピーする場合はバックアップ用だけにしてください。プログラムをそのままの形で、あるいは変更を加えて第三者に販売することは固く禁じられています。

本書には著作権によって保護される内容が含まれています。本書の内容の一部または全部を著作者の許諾なしに複製、改変、および翻訳することは、著作権法下での許可事項を除き、禁止されています。

All rights are reserved.

Restricted Rights Legend.

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company

United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

© Copyright 2002-2004 Hewlett-Packard Development Company, L.P., all rights reserved.

Trademark Notices

OpenView は、Hewlett-Packard Company の米国の登録商標です。

Java™ は、米国 Sun Microsystems, Inc. の商標です。

Oracle® は、米国 Oracle Corporation, Redwood City, California の米国における登録商標です。

UNIX® は、The Open Group の登録商標です。

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

原典

本書は『*HP OpenView Performance Insight Interface Reporting ifEntry Datapipe User Guide Software Version: 2.1 Reporting and Network Solutions 7.0*』 Manufacturing Part No. none (November 2004) を翻訳したものです。

サポート

次の HP OpenView の Web サイトを参照してください。

<http://openview.hp.com/> (英語)

<http://www.hp.com/jp/openview/> (日本語)

これらのサイトには、HP OpenView の提供する製品、サービス、サポートについてのお問い合わせ先や詳細が掲載されています。日本語サポートサイトを表示するには、上記英語版サイトから [**サポート**] をクリックし、[**Software Support Online**] をクリックしてください。サポートサイトでは、お客さまのビジネスの運用に役立つ対話形式の技術サポートツールに手早く効率的にアクセスできます。サポートサイトでは次のことが可能です。

- 関心のあるドキュメントを検索する
- サポートケースを登録 / トラッキングする
- サポート契約を管理する
- HP サポートの問い合わせ先を調べる
- 利用可能なサービスに関する情報を確認する
- 他のお客様とのディスカッションに参加する
- ソフトウェアトレーニングの検索および登録を行う

サポートの多くでは、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。

http://support.openview.hp.com/access_level.jsp

HP Passport ID のご登録は、次の URL で行ってください。

<https://passport.hp.com/hpp2/newuser.do> (英語)

目次

第 1 章	概要	3
	バージョン履歴	3
	追加情報の入手先	3
第 2 章	パッケージのインストール	5
	円滑なインストールのためのガイドライン.....	5
	Interface Reporting ifEntry Datapipe のインストール.....	6
	パッケージの削除	7
第 3 章	データパイプの設定	9
	設定タスク	9
	ポーリンググループの設定	10
	Interface Reporting に付属のフォームへのアクセス.....	11
	プロパティデータのエクスポートとインポート	12
	ポーリングフラグの設定	12
	インタフェース識別子の変更 (再インデックス化のサポート)	15
	複数のポーラー用データパイプの設定	16
第 4 章	グループとポリシー	19
	ポーリンググループ.....	19
	ルールベースのポーリンググループ	20
	ポーリングポリシー.....	21
	IR_map_p	21
第 5 章	ポーリング対象オブジェクトの一覧	23
	表 1 : ifEntry のオブジェクト	23
	表 2 : 拡張 MIB-2 のオブジェクト.....	26
索引		29

概要

Interface Reporting ifEntry Datapipe は、SNMP の管理対象デバイスから MIB-II ifEntry および ifxEntry データを収集します。データの収集に加えて、Interface Reporting ifEntry Datapipe は以下の機能も提供します。

- 再インデックス化のサポート (Interface Discovery Datapipe に基づく)
- 指定インスタンスのポーリング、インスタンス別ポーリングともいう
- 値が常に 32 ビットを超える大容量カウンタ
- ポーリングの設定とネットワークノードマネージャ (NNM) との同期 (任意)

Interface Reporting ifEntry Datapipe は設定不要です。第 3 章に説明されている設定オプションは任意です。

バージョン履歴

バージョン	RNS のバージョンと日付	OVPI バージョン / 拡張機能
1.1	RNS 4.0 - 2003 年 6 月	OVPI 4.5 および 4.6、Sybase のサポート
2.0	RNS 5 - 2004 年 4 月	OVPI 5.0、Oracle のサポート
2.0	RNS 6.0 - 2004 年 8 月	変更なし
2.1	RNS 7.0 - 2004 年 11 月	インタフェースの同期の拡張機能 (OVPI および NNM) - Sybase インタフェースの同期化の拡張機能 (OVPI および NNM) - Oracle 廃棄、エラー、および使用率の割合の計算式に対するマイナー修正

追加情報の入手先

本マニュアルに関連するドキュメントには次のものがあります。

- 『Interface Reporting ifEntry Datapipe Release Statement 2.1』 (英語)
- 『Interface Discovery Datapipe 2.1 ユーザーガイド』
- 『Interface Reporting Report Pack ユーザーガイド』
- 『Frame Relay CPE Datapipe 4.0 ユーザーガイド』
- RNS 7.0 のリリースノート (2004 年 11 月)

主力製品である OVPI のマニュアルと、この製品上で動作するレポート機能についてのマニュアルは、次の Web サイトからダウンロードできます。

<http://www.hp.com/managementsoftware> (英語)

[Support]>[製品マニュアル (英語)] を選択すると、「Product manuals search」ページ (英語) が表示されます。OVPI のマニュアルは [1. Product] で [Performance Insight] を選択すると表示されます。また、レポートパッケージ、データパイプ、および NNM SPI のマニュアルは、[Reporting and Network Solutions] を選択すると表示されます。

日本語のマニュアルは以下の URL から入手できます。

<http://www.jpn.hp.com/doc/manual/openview/index.html>

[Reporting and Network Solutions] の下のマニュアルには、発行年月が示されています。マニュアルが改訂されて新しいものが公開されると、ソフトウェアのバージョン番号が変わらない場合でも、発行年月は変わります。改訂したマニュアルは定期的に公開されるため、このサイトを見て更新されていないかを確認して、最新版でない可能性のある古い PDF を参照するのは避けてください。

パッケージのインストール

本章では、次の項目について説明します。

- 円滑なインストールのためのガイドライン
- パッケージマネージャを使用したパッケージのインストール
- パッケージの削除

円滑なインストールのためのガイドライン

RNS 7.0 CD には、NNM 用のコンポーネントと OVPI のレポートパックが含まれています。OVPI 用のパッケージのインストールを選択した場合は、CD のインストールスクリプトによって CD からすべての OVPI パッケージが抽出され、システムの **Packages** ディレクトリにコピーされます。抽出処理が終わると、インストールスクリプトはパッケージマネージャインストールウィザードを起動するよう要求します。

すでに RNS 7.0 CD から OVPI パッケージを抽出済みの場合は、OVPI に関連するパッケージはシステムにすべてコピーされているので、インストールを開始できます。パッケージマネージャを開始し、画面の指示に従って **Interface Reporting ifEntry Datapipe** をインストールします。パッケージマネージャを使用する前に、次のガイドラインを再確認してください。

ソフトウェアの前提条件

Interface Reporting ifEntry Datapipe 2.1 には、以下の前提条件があります。

- OVPI 5.0
- OVPI 5.0 に提供されているすべてのサービスパック
- Interface Discovery Datapipe 2.1

サービス パックは次のサイトからダウンロードできます。

<http://openview.hp.com/>

Interface Discovery Datapipe を先にインストールすることも、両方のデータパイプを同時にインストールすることもできます。

分散環境

分散システムの場合、インストールはさらに複雑になります。ポーリングを実行するすべての OVPI サーバーに **Interface Reporting ifEntry Datapipe** をインストールするようにしてください。中央サーバーがポーリングを実行しない限り、中央サーバーには **Interface Reporting ifEntry Datapipe** をインストールしないでください。

バージョン 1.1 または 2.0 からのアップグレード

Interface Reporting ifEntry Datapipe にはアップグレード用のパッケージはありません。バージョン 1.1 または 2.0 を実行している場合は、先にパッケージマネージャを起動して現在のバージョンを削除してください。その後、パッケージマネージャを再起動してバージョン 2.1 をインストールします。

Interface Reporting ifEntry Datapipe のインストール

RNS 7.0 CD を挿入し、画面の指示に従い、OVPI ソリューションコンポーネントを抽出します。Windows の場合、[メインメニュー] が自動的に開き、そこに指示手順が表示されます。UNIX の場合、**root** でログインし、CD が自動的にマウントされない場合は CD をマウントし、CD の最上位ディレクトリに移動して **./setup** コマンドを実行します。

OVPI パッケージが抽出され、システムにコピーされると、インストールスクリプトが OVPI を起動し、パッケージマネージャを開始します。CD からパッケージを抽出したところで、パッケージマネージャインストールウィザードが実行中である場合は、この手順を **手順 4** から始めてください。そうでない場合は **手順 1** から始めてください。

- 1 システムにログインします。UNIX システムでは **root** でログインします。
- 2 OVPI Timer を停止し、プロセスが終了するのを待ちます。

Windows: [設定]>[コントロールパネル]>[管理ツール]>[サービス]の順に選択します。

UNIX: **root** でログインし、以下のいずれかを入力します。

HP-UX: **sh /sbin/ovpi_timer stop**

Sun: **sh /etc/init.d/ovpi_timer stop**

- 3 [管理コンソール] から [ツール]、[パッケージマネージャ] の順に選択します。パッケージマネージャの [ようこそ] ウィンドウが開きます。
- 4 [次へ] をクリックします。[OVPI パッケージの場所] ウィンドウが開きます。
- 5 [インストール] をクリックします。デフォルトのインストール先ディレクトリを受け入れるか、必要に応じて別のディレクトリを指定します。
- 6 [次へ] をクリックします。[レポートの配布] ウィンドウが開きます。OVPI ユーザー名とパスワードを入力します。
- 7 [次へ] をクリックします。[パッケージの選択] ウィンドウが開きます。
- 8 次のチェックボックスをクリックします。


Interface_Reporting_ifEntry_Datapipe 2.1

IRifEntry_Sync_Interfaces

- 9 [次へ]をクリックします。[タイプの検出]ウィンドウが開きます。デフォルト値をクリアします。

 **Interface Discovery Datapipe** も同時にインストールする場合、タイプ検出オプションの詳細については『*Interface Discovery Datapipe ユーザーガイド*』を参照してください。

- 10 [次へ]をクリックします。[概要]ウィンドウが開きます。
- 11 [インストール]をクリックします。[インストールプロセス]ウィンドウが開き、インストールが開始されます。インストールが完了すると、パッケージのインストール完了メッセージが表示されます。
- 12 [完了]をクリックして[管理コンソール]に戻ります。

 **Interface Discovery Datapipe** もインストールする場合、**OVPI Timer** を再起動する前に適切なインタフェース識別子 **index-by** グループにデバイスを再割り当てすることが考えられます。この手順については、『*Interface Discovery Datapipe ユーザーガイド*』に説明されています。

- 13 必要があれば、デバイスに適切なインタフェース識別子グループを再割り当てします。
- 14 **OVPI Timer** を再起動します。

Windows: [設定]>[コントロールパネル]>[管理ツール]>[サービス]の順に選択します。

UNIX: root でログインし、以下のいずれかを入力します。

HP-UX: **sh /sbin/ovpi_timer start**

Sun: **sh /etc/init.d/ovpi_timer start**

これで、Interface Reporting ifEntry Datapipe がシステムにインストールされました。

パッケージの削除

Interface Reporting ifEntry Datapipe をアンインストールするには、以下の手順に従います。

- 1 システムにログインします。UNIX システムでは **root** でログインします。
- 2 **OVPI Timer** を停止し、プロセスが終了するのを待ちます。
- 3 パッケージマネージャを開始します。パッケージマネージャのインストールウィザードが開きます。
- 4 画面に表示されるパッケージの削除手順に従い、削除する **Interface Discovery Datapipe** を選択してください。アンインストールが完了すると、パッケージの削除完了メッセージが表示されます。
- 5 [完了]をクリックして[管理コンソール]に戻ります。
- 6 **OVPI Timer** を再起動します。

データパイプの設定

本章では、次の項目について説明します。

- 設定タスクとその実行方法
- ポーリングルールの設定
- **Interface Reporting 4.5** に付属のフォームへのアクセス
- プロパティデータのエクスポートとインポート
- ポーリングフラグの設定
- インタフェース識別子の変更
- 複数のポーラー用データパイプの設定

設定タスク

次の表は、各設定タスクとそれを達成するための方法（複数の場合も有）とを対応付けています。

タスク	手段
ポーリングルールの設定	設定ツール
データフィルタリングモードの設定	インタフェースレポーティングの管理（フォーム）
準備モードの設定	インタフェースレポーティングの管理（フォーム）
メッセージレベルの設定	インタフェースレポーティングの管理（フォーム）
プロパティデータのエクスポートとインポート	IR 設定ツール
ポーリングフラグの設定	<ol style="list-style-type: none"> 1 インタフェースポーリングフラグの管理（フォーム） 2 ポーリングポリシーマネージャ 3 IR Report Pack のプロパティのインポート 4 NNM とのポーリングの同期 5 カスタム SQL スクリプト

タスク	手段
プロトコルの監視の無効化	プロトコルデフォルトの変更(フォーム)
インタフェース識別子の変更	<ol style="list-style-type: none"> 1 Interface Discovery Datapipe 2 ポーリングポリシーマネージャ
複数のポーラー用データパイプの設定	<ol style="list-style-type: none"> 1 OVPI group_manager 2 OVPI collection_manager 3 ポーリングポリシーマネージャ

ポーリングルールの設定

ポーリングは、以下の 5 つのルールのいずれかに従って行われます。

(1) Poll All Active Interfaces(すべてのアクティブなインタフェースをポーリングする)

ifAdminStatus が Active に設定されているすべてのインタフェースをポーリングします。

(2) Poll All Interfaces(すべてのインタフェースをポーリングする)

ifAdminStatus またはプロトコルに関係なく、すべてのインタフェースをポーリングします(これは、Interface Reporting ifEntry Datapipe がインストールされたときのデフォルトのルールです)。

(3) Poll All Active Interfaces with Monitored Protocols(監視対象プロトコルが設定されたすべてのアクティブなインタフェースをポーリングする)

デフォルトのルールです。ifAdminStatus が Active に設定され、KIR_protocol_config テーブルでプロトコル(ifType)が監視対象として設定されているすべてのインタフェースをポーリングします。監視の不要なプロトコルについては監視を無効にできます。



デフォルトのルールは「Poll All Active Interfaces with Monitored Protocols」(監視対象プロトコルが設定されたすべてのアクティブなインタフェースをポーリングする)です。

(4) Poll All Interfaces with Monitored Protocols(監視対象プロトコルが設定されたすべてのインタフェースをポーリングする)

KIR_protocol_config テーブルでプロトコル(ifType)が監視対象として設定されているすべてのインタフェースをポーリングします。監視の不要なプロトコルについては監視を無効にできます。

(5) Poll Flagged Interfaces Only(フラグが設定されたインタフェースのみをポーリングする)

ポーリングフラグが設定されたインタフェースのみをポーリングします。この設定では、個別のインタフェースを収集用に準備できます。この設定を使用すると、ポーリングを制御してNNMの「非管理対象インタフェース」機能と統合させることができます。

ポーリングルールの設定を行うには、次の手順に従います。

- 1 システムにログインします。UNIX の場合は、`root` でログインします。
- 2 コマンド行から IR 設定ツールを起動します。

UNIX:

```
$DPIPE_HOME/bin/perl $DPIPE_HOME/scripts/IR_Config.pl
```

Windows:

```
%DPIPE_HOME%\bin\perl %DPIPE_HOME%\scripts\IR_Config.pl
```

[IR Configuration Tool] メインメニューが表示されます。

- 3 オプション 1 の [Configure Polling] を選択し、[Enter] を押します。
[Configure Polling] メニューが開き、現在のポーリングルールと、そのルールが最後に変更された日時が表示されます。
- 4 必要なポーリングルールのオプションを選択し、[Enter] を押します。
ポーリングルールを実装するデータベースビューが再生成され、メッセージが表示された後、再び [Set Polling Rule] メニューが開き、新しいポーリングルールが表示されます。
- 5 [E] を選択して [Enter] を押し、前のメニューに戻ります。
IR 設定ツールのメインメニューが開きます。
- 6 オプションを選択するか、[E] を選択し、[Enter] を押して終了します。



「Poll Flagged Interfaces Only」(フラグの設定されたインタフェースのみをポーリングする)ルールを適用する場合、ポーリング対象のインタフェースにもポーリングフラグを設定する必要があります。

Interface Reporting に付属のフォームへのアクセス

Interface Reporting4.5 に付属のフォームにアクセスするには、次の手順に従います。

- 1 [管理コンソール] を起動します。
- 2 [オブジェクト] アイコンをクリックします。[オブジェクト/プロパティの管理] ウィンドウが開きます。
- 3 デバイスを選択します。

[オブジェクト固有のタスク] の下に以下のフォームが表示されます。

- プロトコルデフォルトの変更



デフォルトでは、すべてのプロトコル (ifTypes) が監視され、すべてのプロトコルのデータが収集されます。特定のプロトコルに関心がない場合は、プロトコルデフォルトの変更フォームを使用して収集およびレポートを無効にすることができます。

- インタフェースのプロパティの変更

- インタフェースのカスタマの変更
- インタフェースの説明の変更
- インタフェースポーリングフラグの管理

[**全般的なタスク**] の下に以下のフォームが表示されます。

- インタフェースレポーティングの管理

フォームの詳細については、『*Interface Reporting Report Pack ユーザーガイド*』の第6章「変更フォームの使用」を参照してください。

プロパティデータのエクスポートとインポート

IR 設定ツールを使用すると、容易に Interface Reporting Report Pack から既存のプロパティデータを抽出し、新しいプロパティデータを Interface Reporting Report Pack にインポートすることができます。IR 設定ツールを使用してプロパティデータをエクスポートまたはインポートすると、データファイルは、以下の、Interface Reporting Report Pack が所有するプロパティデータのデフォルトディレクトリとの間でエクスポートまたはインポートされます。

`$DPIPE_HOME/data/PropertyData`

プロパティデータをエクスポートまたはインポートするには、以下の手順に従います。

- 1 システムにログインします。UNIX の場合は、`root` でログインします。
- 2 コマンド行から IR 設定ツールを起動します。

UNIX:

```
$DPIPE_HOME/bin/perl $DPIPE_HOME/scripts/IR_Config.pl
```

Windows:

```
%DPIPE_HOME%\bin\perl %DPIPE_HOME%\scripts\IR_Config.pl
```

IR 設定ツールのメインメニューが開きます。

- 3 オプション 2 の [Export or Import Interface Reporting Property Data] を選択し、[Enter] を押します。

[Property Export and Import] メニューが開きます。

- 4 以下のいずれかを実行します。
エクスポートするには [1. Export Interface Reporting Property Data] を選択します。
インポートするには [2. Import Interface Reporting Property Data] を選択します。

- 5 終了したら、[E] を選択して [Enter] を押し、前のメニューに戻ります。

[IR Configuration Tool] メインメニューが表示されます。

- 6 オプションを選択するか、[E] を選択し、[Enter] を押し、終了します。

ポーリングフラグの設定

ポーリングフラグを設定するには、以下のいずれかの方法で行います。

- インタフェースポーリングフラグの管理 (フォーム)
- ポーリングポリシーマネージャ
- Interface Reporting Report Pack のプロパティのインポート
- NNM の「非管理対象インタフェース」ポーリング設定との同期



この方式によるポーリングフラグの設定は、Sybase 上の OVPI にのみ適用されます。

- カスタム SQL スクリプトの作成

ポーリングポリシーマネージャを使用したポーリングフラグの設定

「Poll Flagged Interfaces Only」(フラグが設定されたインタフェースのみをポーリングする)ルールが有効な場合、Interface Reporting ifEntry Datapipe はメイン インタフェースプロパティテーブル、K_IFEntry_Disc.poll_flag_ifEntry でポーリングフラグが設定されているインタフェースからのみ統計情報を収集します。ポーリングポリシーマネージャを使用してポーリングフラグを変更するには、以下の手順を実行します。

- 1 どのデバイスのどのインタフェースをポーリングするか、またはポーリングしないかを確定します。
- 2 システムにログインします。UNIX の場合は、root でログインします。
- 3 ポーリングポリシーマネージャを起動します。
- 4 [編集]>[説明...]の順に選択します。[説明の編集]ウィンドウが開きます。
- 5 Property Group で「K_IFEntry_Disc」を選択し、[Rule:]の横にある[Edit...]をクリックします。[ルールの作成]ウィンドウが開きます。
- 6 ルールを選択または入力します。たとえば、高速のインタフェースのみを返すには、「interface_speed >= 100000000」を使って[OK]をクリックします。
- 7 [結果のインスタンスの表示]をクリックします。インスタンスが下のペインに表示されます。
- 8 返されたリストから、ポーリング対象のすべてのインタフェースを選択し、[インスタンス値の編集...]をクリックします。[インスタンスの編集]ウィンドウが開きます。
- 9 [poll_flag_ifEntry]列を選択し、1の値を入力して[Aply]をクリックし、[閉じる]をクリックして[説明の編集]ウィンドウに戻ります。
- 10 [結果のインスタンスの表示]をクリックします。更新されたインスタンスが下のペインに表示されます。
- 11 [閉じる]をクリックします。
- 12 「Poll Flagged Interfaces Only」(フラグの設定されたインタフェースのみをポーリングする)ルールがまだ有効化されていない場合は、ここで有効にします。

プロパティインポートを使用したポーリングフラグの設定

「Poll Flagged Interfaces Only」（フラグが設定されたインタフェースのみをポーリングする）ルールが有効な場合、**Interface Reporting ifEntry Datapipe** はメイン インタフェースプロパティテーブルでポーリングフラグが設定されているインタフェースからのみ統計情報を収集します。**[Interface Reporting Property Import]** を使用してポーリングフラグを設定するには、以下の手順に従います。

▶ 準備されたインタフェースのデータのみが収集されるようにデータフィルタリングモードが有効になっている場合は、プロパティインポートファイルに含まれている各インタフェースに対して、ポーリングフラグが自動的に設定されます。

- 1 どのデバイスのどのインタフェースをポーリングするか、またはポーリングしないかを確定します。
- 2 システムにログインします。UNIX の場合は、**root** でログインします。
- 3 ポーリング対象のインタフェースによる新しい **Interface Reporting** プロパティインポートファイルを作成するか、インタフェースがすでに検出済みの場合は、プロパティファイルをエクスポートして必要なインタフェースのみを含むように編集します。
- 4 インタフェースレポーティングの管理フォームを使用して、データフィルタリングモードが有効になっているかどうかを調べます。データフィルタリングモードがまだ有効化されていない場合は、ここで有効にします。
- 5 手順 3 で生成した **Interface Reporting** プロパティデータをインポートします。
- 6 「Poll Flagged Interfaces Only」（フラグの設定されたインタフェースのみをポーリングする）ルールがまだ有効化されていない場合は、ここで有効にします。

NNM との同期化によるポーリングフラグの設定

▶ この方式によるポーリングフラグの設定は、Sybase 上の OVPI にのみ適用されます。

「Poll Flagged Interfaces Only」（フラグが設定されたインタフェースのみをポーリングする）ルールが有効な場合、**Interface Reporting ifEntry Datapipe** はメイン インタフェースプロパティテーブルでポーリングフラグが設定されているインタフェースからのみ統計情報を収集します。このフラグは、NNM の「管理対象でないインタフェース」のポーリング設定と同期することによって設定できます。NNM では、ユーザーは、ユーザーインタフェースを使う方法と、これに使用する目的で設計された、特殊なアドオン方式のルールをベースにしたツールを使う方法の、2 種類の方法で非管理対象インタフェースを定義できます。

IRifEntry Sync Interfaces パッケージは、NNM から管理対象および非管理対象インタフェースリストをアップロードし、管理対象インタフェースについてはポーリングフラグを設定し、非管理対象インタフェースについてはフラグをクリアするように設計されています。このパッケージは、**Integration Module** の **Device Sync** パッケージに依存しています。これらのパッケージのインストールに関する詳細は、『*NNM/Performance Insight Integration Module 2.0 User Guide*』（英語）を参照してください。

IRifEntry Sync Interfaces パッケージをインストールすると、次のイベントが発生します。

- 管理対象インタフェースおよび非管理対象インタフェースの完全なリストが NNM からダウンロードされる。
- インタフェースポーリングフラグが (NNM の設定に基づいて) 設定される。
- 以下の項目に対して OVPI Timer エントリが設定される。
 - 過去 24 時間以内に管理ステータスが変更されたすべてのインタフェースの日単位の同期
 - 月 1 回の完全な再同期

NNM との同期によってポーリングフラグを設定するには、次の手順に従います。

- 1 Integration Module に付属のパッケージがまだ NNM および OVPI にインストールされていない場合は、それらをインストールします。
- 2 IRifEntry Sync Interfaces パッケージがまだインストールされていない場合は、ここでインストールします。第 2 章「パッケージのインストール」に説明されている手順に従います。
- 3 「Poll Flagged Interfaces Only」(フラグの設定されたインタフェースのみをポーリングする) ルールがまだ有効化されていない場合は、ここで有効にします。

カスタム SQL スクリプトの作成

ポーリングの設定が準備管理システムまたは設定管理システムで保持されている場合は、カスタム SQL スクリプトを開発してポーリングフラグを設定します (K_IFEntry_Disc.poll_flag_ifEntry)。

インタフェース識別子の変更 (再インデックス化のサポート)

Interface Discovery Datapipe では、以下の代替識別子を使用できます。

- ifDescr — index-by-ifDescr ポーリンググループによりサポート
- ifName — index-by-ifName ポーリンググループによりサポート
- ifAlias — index-by-ifAlias ポーリンググループによりサポート

再インデックス化を有効にするには、デバイスで一意的、固定された代替識別子 (1 つだけの代替識別子) を選択します。



デバイスを複数のグループに割り当てることは**できません**。デバイスあたり 1 つの代替識別子のみが許可されています。

Interface Discovery Datapipe に、特定の種類のデバイスに自動的に代替インタフェース識別子を割り当てるベンダー固有の OVPI タイプ検出ファイルが含まれている場合は、一部の選択作業はすでに完了しています。次の表に、デバイスタイプと事前に割り当てられた識別子を示します。

デバイスタイプ	OVPI タイプ検出によって割り当てられた識別子
Cisco Catalyst スイッチ	ifName
Cisco ルーター	ifDescr
HP ProCurve スイッチ (800T を除く)	ifName

index-by-ifIndex グループはデフォルトのグループです。GENMIB2IF_Datapipe グループに属しており、index-by-ifAlias、index-by-ifDescr、または index-by-ifName グループには属さないすべてのデバイスがこのグループに含まれます。

インタフェース識別子を変更するには、以下の手順に従います。

- 1 各デバイスタイプごとに、インタフェース識別子を選択します。
- 2 そのタイプの各デバイスを適切な index-by グループに割り当てます。
 - a ポーリングポリシーマネージャを起動します。
 - b [編集]>[ポーリンググループ ...]の順に選択します。
 - c 上のペインで[同じタイプのすべてのノード]を選択し、下のペインで必要な index-by グループを選択します。
 - d [編集]をクリックします。
 - e このグループに割り当てられるノードを左側のペインで選択し、[>>]をクリックしてグループの中に移動します。完了したら[OK]をクリックします。
- 3 必要に応じて、その他のデバイスタイプおよび index-by グループについても上記の手順を繰り返します。

デバイスを index-by グループに割り当てると、識別子が一意でない場合は、識別子が重複していることを示す次のようなメッセージが trendlog に挿入されます。

```
bcp_gateway, ,WARNING,1984,2076,0,"the object [testhost.test.hp.com_AT3/0] is duplicate for data table rv_ifentry_disc_ifname"
```

複数のポーラー用データパイプの設定

複数のポーラーで Interface Reporting ifEntry Datapipe を使用するには追加の設定が必要です。標準のインストールでは、1つのポーラーに対して8つのグループと4つのポーリングポリシーが設定されます。複数のポーラーで使用するには、追加のポーラーごとに追加のグループおよびポーリングポリシーのセットを設定する必要があります。

データパイプパッケージには、2つのポーラー用の追加のグループおよび収集ポリシーファイルが含まれています。1つのデータベースに対して3つ以上のポーラーのデータパイプを設定する必要がある場合には、以下の手順に従って、グループファイルとポリシーの追加セットを生成してください。

- 1 OVPI 管理者 (trendadm) としてログインします。
- 2 Interface Reporting ifEntry Datapipe パッケージのディレクトリに移動します。
- 3 次のコマンドを実行します。

```
trend_proc -f create_groups_multi_pollers.pro
```

- 4 次のコマンドを実行します。

```
collection_manager -import -file IRifEntry_Collection_Policies_multi_pollers
```

- 5 ポーリングポリシーマネージャを起動し、[編集]>[データパイプのインストール...]を選択します。
- 6 [作成...]をクリックして、新しいポーラーを作成します。
- 7 各ポーラーのホスト名とIPアドレス(任意)を追加し、[OK]をクリックします。
- 8 すべてのポーラーを追加したら、[閉じる]をクリックして[データパイプのインストールの編集]ウィンドウを終了します。
- 9 左のパネルの[すべて]をクリックしてすべてのポーリングポリシーを表示します。
- 10 元のif*_Collection ポリシー(名前の末尾に_poller1 または_poller2 がないもの)を削除します。
- 11 新しいif*_Collection_poller1 ポリシーをダブルクリックして変更します。
[ポーリングの割り当て先]設定を poller1 のホスト名に変更します。
- 12 新しいif*_Collection_poller2 ポリシーをダブルクリックして変更します。
[ポーリングの割り当て先]設定を poller2 のホスト名に変更します。

この時点で、15分単位のポリシーは下表のポリシーのようになります。

ポリシー名	収集されるデータのテーブルエイリアス名	収集元のグループ名	ポーリング元
ifAlias_Collection_poller1	xV_IRifEntry_ifAlias	ifAlias-poller1	poller1 のホスト名
ifDescr_Collection_poller1	xV_IRifEntry_ifDescr	ifDescr-poller1	poller1 のホスト名
ifIndex_Collection_poller1	xV_IRifEntry_ifIndex	ifIndex-poller1	poller1 のホスト名
ifName_Collection_poller1	xV_IRifEntry_ifName	ifName-poller1	poller1 のホスト名
ifAlias_Collection_poller2	xV_IRifEntry_ifAlias	ifAlias-poller2	poller2 のホスト名
ifDescr_Collection_poller2	xV_IRifEntry_ifDescr	ifDescr-poller2	poller2 のホスト名
ifIndex_Collection_poller2	xV_IRifEntry_ifIndex	ifIndex-poller2	poller2 のホスト名
ifName_Collection_poller2	xV_IRifEntry_ifName	ifName-poller2	poller2 のホスト名

- 13 上のペインで [編集]>[ポーリンググループ ...]>[同じビュー内にあるすべてのノード] の順に選択し、次に下のペインで [poller1] を選択します。poller1 によって収集されるノードを [poller1] ビューに割り当てます。
- 14 上のペインで [編集]>[ポーリンググループ ...]>[同じビュー内にあるすべてのノード] の順に選択し、次に下のペインで [poller2] を選択します。poller2 によって収集されるノードを [poller2] ビューに割り当てます。

グループとポリシー

本章では、次の項目について説明します。

- ポーリンググループ
- ルールベースのポーリンググループ
- ポーリングポリシー
- IR_map_p ストアドプロシージャ

ポーリンググループ

Interface Reporting `ifEntry Datapipe` は、Interface Discovery `Datapipe` と同じ OVPI ポーリング「タイプ」グループを使用します。ポーリンググループは、再インデックス化のサポートを有効にします。これらのポーリンググループは `Interface Discovery Datapipe` のインストール時に作成されます。

ポーリンググループ	説明
<code>GENMIB2IF_Datapipe</code>	MIB-II をサポートするすべてのノードの最上位レベルのグループです。
<code>index-by-ifAlias</code>	<code>GENMIB2IF_Datapipe</code> グループのサブセット。識別子として <code>ifAlias</code> を使用するすべてのノードを含みます。
<code>index-by-ifDescr</code>	<code>GENMIB2IF_Datapipe</code> グループのサブセット。識別子として <code>ifDescr</code> を使用するすべてのノードを含みます。
<code>index-by-ifIndex</code>	<code>GENMIB2IF_Datapipe</code> グループのサブセット。 <code>index-by-ifAlias</code> 、 <code>index-by-ifDescr</code> 、または <code>index-by-ifName</code> グループに属さないすべてのデバイスがこのグループに含まれます。デバイスが別のグループに割り当てられない限り、デバイスのインタフェース識別子のデフォルト値は <code>ifIndex</code> です。このグループの編集は必要ありません。
<code>index-by-ifName</code>	<code>GENMIB2IF_Datapipe</code> グループのサブセット。識別子として <code>ifName</code> を使用するすべてのノードを含みます。

index-by-ifIndex グループはデフォルトのグループです。このグループは、ポーリングの制御よりも、インタフェース識別子として **ifIndex** を使用するデバイスを表示します。このグループは 1 時間に 1 回、自動的に再生成されるため、デバイスが別の **index-by** グループに再割り当てされても、即座に **index-by-ifIndex** から消去されない場合があります。ただし、**index-by-ifIndex** のデータの収集は **GENMIB2IF_Datapipe** グループ内のデバイスを基にしているため、デバイスは新しい **index-by** グループからのみポーリングされます。

ルールベースのポーリンググループ

5 つの基本のポーリンググループに加えて、**Interface Reporting ifEntry Datapipe** は、基本グループから派生していて、しかもルールベースのポーリングのルールも含む 4 つのグループを追加します。

グループ名	データベースビュー	説明
ifAlias-all	KV_IFEntry_by_ifAlias	ポーリングルールが割り当てられた index-by-ifAlias グループから派生。
ifDescr-all	KV_IFEntry_by_ifDescr	ポーリングルールが割り当てられた index-by-ifDescr グループから派生。
ifIndex-all	KV_IFEntry_by_ifIndex	ポーリングルールが割り当てられた、 GENMIB2IF_Datapipe グループに属していて、 index-by-ifAlias 、 index-by-ifDescr 、または index-by-ifName に属さないデバイスから派生。
ifName-all	KV_IFEntry_by_ifName	ポーリングルールが割り当てられた index-by-ifName グループから派生。

各グループでポーリングされるインタフェースの一覧を表示するには、**SQL** を使用して上記のテーブルのデータベースビューから選択します。

ポーリングポリシー

Interface Reporting ifEntry Datapipe の標準のインストールでは、4つの15分単位のポーリングポリシーのセットが、各 index-by グループごとに1つずつ有効になります。複数ポーラーのサポートを有効にする必要がない限り、これらのポリシーを変更する必要はありません。

ポリシー名	収集されるデータのテーブルエイリアス名	収集元のグループ名
ifAlias_Collection	xV_IRifEntry_ifAlias	ifAlias-all
ifDescr_Collection	xV_IRifEntry_ifDescr	ifDescr-all
ifIndex_Collection	xV_IRifEntry_ifIndex	ifIndex-all
ifName_Collection	xV_IRifEntry_ifName	ifName-all

Interface Reporting ifEntry Datapipe は、Interface Discovery Datapipe プロパティテーブル K_IFEntry_Disc に poll_flag_ifEntry という名前のフラグ列を追加します。このフラグは、指定インスタンスポーリングの制御用に準備できます。

IR_map_p

Interface Reporting ifEntry Datapipe は、IR_map_p というストアドプロシージャを使用して収集したデータを Interface Reporting Report Pack に転送します。IR_map_p は、raw-to-rate 変換の最後に呼び出されます。IR_map_p および変換プロシージャは、上記の4つのポーリングポリシーの最後に実行されます。実行中、これらのプロシージャはデータが失われないようにするためテーブルをロックします。その結果、raw-to-rate ストアドプロシージャの1つまたは複数のインスタンスで、すでにインスタンスが実行中であることを示すロックエラーが生成されることがよくあります。エラーメッセージの例を以下に示します。

```
"Msg 20003, Level 16, State 1:"
```

```
"Server 'MYOVPI_SYBASE', Procedure 'dsi_locks_p', Line 267:"
```

```
"Unable to lock the object 'x_IRifEntryTOR_IRifEntry28_p'."
```

```
"Error: raw2rate procedure failed for table xV_IRifEntry_ifindex, database MYOVPI_SYBASE. Exit code=1"
```

IR_map_p は、Interface Reporting Report Pack に転送された行数を示す標準の情報メッセージも生成します。次に、例を示します。

```
IR_map_p: Info: 20420 polled rows were mapped to SRIRdevPorts in 18 seconds.
```


ポーリング対象オブジェクトの一覧

以下の表に、ポーリング対象オブジェクトの名前、識別子、説明を示します。表 1(下の表)は、オブジェクトの一覧を、ifEntry MIB テーブルのオブジェクト ID 別に示しています。表 2(26 ページ)は、オブジェクトの一覧を、拡張 MIB-2 テーブル (ifXTable ともいう) のオブジェクト ID 別に示しています。

表 1 : ifEntry のオブジェクト

MIB オブジェクト名	オブジェクト ID	説明
ifIndex	1.3.6.1.2.1.2.2.1.1	1 からの ifNumber の値までの、各インタフェースの一意的値。各インタフェースの値は、少なくとも、エンティティのネットワーク管理システムの 1 回の再初期化から次に再初期化するまでは一定でなければなりません。
ifDescr	1.3.6.1.2.1.2.2.1.2	インタフェースに関する情報を含むテキスト文字列。この文字列には、ハードウェアインタフェースの製造元メーカー名、製品名、およびバージョンが含まれます。
ifType	1.3.6.1.2.1.2.2.1.3	プロトコルスタックにおいてネットワーク層の直下にある物理/リンクプロトコルに従って区別される、インタフェースの種類。
ifMtu	1.3.6.1.2.1.2.2.1.4	インタフェース上で送受信可能なデータグラムの最大サイズ。オクテット単位で指定。ネットワークデータグラムの送信で使用されるインスタンスの場合は、インタフェースで送信できるネットワークデータグラムの最大サイズです。
ifSpeed	1.3.6.1.2.1.2.2.1.5	インタフェースの現在の帯域幅 (ビット / 秒) の予測値。帯域幅が変動しない場合や、正確な予測が不可能な場合は、このオブジェクトには公称帯域幅が含まれます。

表 1 : ifEntry のオブジェクト

MIB オブジェクト名	オブジェクト ID	説明
ifPhysAddress	1.3.6.1.2.1.2.2.1.6	プロトコルスタックにおいてネットワーク層の直下にあるプロトコル層のインタフェースのアドレス。そのようなアドレスがないインタフェース(シリアルラインなど)の場合、このオブジェクトには長さがゼロのオクテット文字列が含まれます。
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	求められるインタフェースの状態。テスト状態では、動作パケットを渡すことはできません。状態には、動作(1)、停止(2)、テスト(3)があります。
ifOperStatus	1.3.6.1.2.1.2.2.1.8	インタフェースの現在の動作状態。テスト状態では、動作パケットを渡すことはできません。状態には、動作(1)、停止(2)、テスト(3)があります。
ifLastChange	1.3.6.1.2.1.2.2.1.9	インタフェースが現在の動作状態になった時点の sysUpTime の値。ローカルネットワーク管理サブシステムの初期化以前から、動作状態が変わっていない場合はこのオブジェクトにゼロの値が入ります。
IfInOctets	1.3.6.1.2.1.2.2.1.10	インタフェースで受信された総オクテット数(フレーミング文字を含む)。
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	上位層のプロトコルに送信されたサブネットワークユニキャストパケットの数。
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	上位層のプロトコルに送信された非ユニキャスト(サブネットワークブロードキャストまたはサブネットワークマルチキャスト)パケットの数。
ifInDiscards	1.3.6.1.2.1.2.2.1.13	上位層のプロトコルに送信されないようにするため、エラーが検出されていなくても廃棄するように選択された着信パケットの数。このようなパケットの廃棄の理由の1つとして、バッファスペースの解放があります。
ifInErrors	1.3.6.1.2.1.2.2.1.14	上位層のプロトコルへの送信を妨げるエラーが含まれていた着信パケットの数。
ifUnknownProtos	1.3.6.1.2.1.2.2.1.15	不明なプロトコルまたはサポートされていないプロトコルであるために廃棄された、インタフェース経由で受信されたパケットの数。
ifOutOctets	1.3.6.1.2.1.2.2.1.16	インタフェースから送信された総オクテット数(フレーミング文字を含む)。
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	上位のプロトコルがサブネットワークユニキャストへの送信を要求したパケットの総数(廃棄されたり送信されなかったものも含む)。

MIB オブジェクト名	オブジェクト ID	説明
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	上位のプロトコルが非ユニキャスト（サブネットワークブロードキャストまたはサブネットワークマルチキャスト）アドレスへの送信を要求したパケットの総数（廃棄されたり送信されなかったものも含む）。
ifOutDiscards	1.3.6.1.2.1.2.2.1.19	送信を妨げるようなエラーが検出されていなくても廃棄するよう選択された送信パケットの数。このようなパケットの廃棄の理由の1つとして、バッファスペースの解放があります。
ifOutErrors	1.3.6.1.2.1.2.2.1.20	エラーのために送信できなかった送信パケットの数。
ifOutQLen	1.3.6.1.2.1.2.2.1.21	送信パケットキューの長さ（パケット単位）。
ifSpecific	1.3.6.1.2.1.2.2.1.22	インタフェースを認識するために使用されているメディア固有の、MIB 定義の参照先。たとえば、インタフェースがイーサネットによって認識されている場合、このオブジェクトの値はイーサネット固有のオブジェクトを定義しているドキュメントを参照します。この情報が存在しない場合、この値は構文的に有効なオブジェクト識別子である、 OBJECT IDENTIFIER { 0 0 } に設定する必要があります。これは構文的に有効なオブジェクト識別子であり、 ASN.1 および BER に適合する実装はすべて、この値を生成および認識する必要があります。

表 2 : 拡張 MIB-2 のオブジェクト

MIB オブジェクト名	オブジェクト ID	説明
ifName	1.3.6.1.2.1.31.1.1.1.1	<p>インタフェースのテキスト名。このオブジェクトの値は、ローカルデバイスによって割り当てられたインタフェース名であり、デバイスのコンソールからのコマンド入力での使用に適している必要があります。デバイスのインタフェース命名構文によって、「le0」などのテキスト名の場合と、1 などの単純なポート番号の場合があります。ifTable の複数のエントリがまとまって、デバイスによって命名された 1 つのインタフェースを表している場合は、それぞれのエントリの ifName の値が同じになります。</p> <p>ローカル名がない場合、またはこのオブジェクトが他に適用できない場合、オブジェクトには長さが 0 の文字列が入ります。</p>
ifAlias	1.3.6.1.2.1.31.1.1.1.18	<p>このオブジェクトはネットワーク管理者によって指定されたインタフェースのエイリアス名であり、持続的に利用できるインタフェースへの「ハンドル」として機能します。</p> <p>インタフェースが最初にインスタンス化された時点では、そのインタフェースに関連付けられている ifAlias の値は長さがゼロの文字列です。ネットワーク管理設定操作を通じて ifAlias のインスタンスに値が書き込まれると、ネットワーク管理システムが再初期化またはリブートされても、エージェントはそのインタフェースが次にインスタンス化されるまでは同じインタフェースに関連付けられている ifAlias インスタンスに提供された値を保持する必要があります。</p> <p>ネットワーク管理者が WAN インタフェース用にこのオブジェクトに格納する値には、(通信事業者の)回線番号やインタフェースの識別子などがあります。エージェントによっては特定の ifType の値を持つインタフェースに対してのみ、書き込みアクセスをサポートしている場合があります。このオブジェクトに対して書き込みアクセスをサポートしているエージェントは、安定した記憶領域に値を保存する必要がありますが、すでに他のインタフェースの現在の値によって占有されている記憶領域の量によっては、新しい値の長さが制限される場合があります。</p>
ifHCInOctets	1.3.6.1.2.1.31.1.1.1.6	<p>インタフェースで受信された総オクテット数(フレーミング文字を含む)。このオブジェクトは ifInOctets の 64 ビットバージョンです。</p>

MIB オブジェクト名	オブジェクト ID	説明
ifHCInUcastPkts	1.3.6.1.2.1.31.1.1.1.7	このサブレイヤーのマルチキャストアドレスまたはブロードキャストアドレスにアドレス指定されなかった、このサブレイヤーから上位の(サブ)レイヤーに送信されたパケットの数。このオブジェクトは ifInUcastPkts の 64 ビットバージョンです。
ifHCInMulticastPkts	1.3.6.1.2.1.31.1.1.1.8	このサブレイヤーのマルチキャストアドレスにアドレス指定された、このサブレイヤーから上位の(サブ)レイヤーに送信されたパケットの数。 MAC レイヤープロトコルの場合、これには Group アドレスと Functional アドレスの両方が含まれます。このオブジェクトは ifInMulticastPkts の 64 ビットバージョンです。
ifHCInBroadcastPkts	1.3.6.1.2.1.31.1.1.1.9	このサブレイヤーのブロードキャストアドレスにアドレス指定された、このサブレイヤーから上位の(サブ)レイヤーに送信されたパケットの数。このオブジェクトは ifInBroadcastPkts の 64 ビットバージョンです。
ifHCOutOctets	1.3.6.1.2.1.31.1.1.1.10	インタフェースから送信された総オクテット数(フレーミング文字を含む)。このオブジェクトは ifOutOctets の 64 ビットバージョンです。
ifHCOutUcastPkts	1.3.6.1.2.1.31.1.1.1.11	上位のプロトコルがサブネットワークユニキャストへの送信を要求したパケットの総数(廃棄されたり送信されなかったものも含む)。このオブジェクトは ifOutUcastPkts の 64 ビットバージョンです。
ifHCOutMulticastPkts	1.3.6.1.2.1.31.1.1.1.12	このサブレイヤーのマルチキャストアドレスにアドレス指定された、上位のプロトコルが送信を要求したパケットの総数(廃棄されたり送信されなかったものも含む)。 MAC レイヤープロトコルの場合、これには Group アドレスと Functional アドレスの両方が含まれます。このオブジェクトは ifOutMulticastPkts の 64 ビットバージョンです。
ifHCOutBroadcastPkts	1.3.6.1.2.1.31.1.1.1.13	このサブレイヤーのブロードキャストアドレスにアドレス指定された、上位のプロトコルが送信を要求したパケットの総数(廃棄されたり送信されなかったものも含む)。このオブジェクトは ifOutBroadcastPkts の 64 ビットバージョンです。

C

Cisco Catalyst スイッチ、代替識別子, 16
Cisco ルーター、代替識別子, 16

G

GENMIB2IF_Datapipe ポーリンググループ, 19

H

HP ProCurve スイッチ、代替識別子, 16

I

ifAdminStatus オブジェクトの説明, 24
ifAlias オブジェクトの説明, 26
ifDescr オブジェクトの説明, 23
ifEntry MIB テーブル、オブジェクト, 23
ifHCInBroadcastPkts オブジェクトの説明, 27
ifHCInMulticastPkts オブジェクトの説明, 27
ifHCInOctets オブジェクトの説明, 26
ifHCInUcastPkts オブジェクトの説明, 27
ifHCOutBroadcastPkts オブジェクトの説明, 27
ifHCOutMulticastPkts オブジェクトの説明, 27
ifHCOutOctets オブジェクトの説明, 27
ifHCOutUcastPkts オブジェクトの説明, 27
ifIndex オブジェクトの説明, 23
ifInDiscards オブジェクトの説明, 24
ifInErrors オブジェクトの説明, 24
ifInNUcastPkts オブジェクトの説明, 24
IfInOctets オブジェクトの説明, 24
ifInUcastPkts オブジェクトの説明, 24
ifLastChange オブジェクトの説明, 24
ifMtu オブジェクトの説明, 23
ifName オブジェクトの説明, 26
ifOperStatus オブジェクトの説明, 24
ifOutDiscards オブジェクトの説明, 25
ifOutErrors オブジェクトの説明, 25
ifOutNUcastPkts オブジェクトの説明, 25
ifOutOctets オブジェクトの説明, 24
ifOutQLen オブジェクトの説明, 25
ifOutUcastPkts オブジェクトの説明, 24
ifPhysAddress オブジェクトの説明, 24
ifSpecific オブジェクトの説明, 25
ifSpeed オブジェクトの説明, 23
ifType オブジェクトの説明, 23
ifUnknownProtos オブジェクトの説明, 24
index-by-ifAlias ポーリンググループ, 19
index-by-ifDescr ポーリンググループ, 19
index-by-ifindex ポーリンググループ, 19, 20
index-by-ifName ポーリンググループ, 19
Interface Reporting Property Import
 ポーリングフラグの設定, 14
IR_map_p ストアドプロシージャ, 21
IRifEntry Sync Interfaces パッケージ, 14

K

K_IFEntry_Disc プロパティテーブル, 21

M

MIB オブジェクト、一覧, 23

O

OVPI Timer, 7

P

poll_flag_ifEntry, 21

ProCurve スイッチ、代替識別子, 16

「Product Manuals Search」のページ, 4

S

SQL スクリプト、ポーリングフラグの設定用, 15

あ

インストール

指示手順, 6

分散環境, 6

オブジェクト

ifEntry MIB テーブル, 23

拡張 MIB-2 テーブル, 26

ポーリング対象, 23

か

拡張 MIB-2 テーブル、オブジェクト, 26

さ

再インデックス化のサポート, 15

識別子、代替, 15

指定インスタンスポーリング, 21

製品ドキュメント, 3

設定

複数のポーラー, 16

ポーリングルール, 11

前提条件、ソフトウェア, 5

た

代替識別子, 15

代替識別子、変更, 15

タイプ検出ファイル, 16

データパイプ

インストール, 6

削除, 7

データパイプのアップグレード, 6

データパイプのアンインストール, 7

な

ネットワークノードマネージャ (NNM)

ポーリングフラグの設定, 14

は

非管理対象インタフェースのポーリング設定, 14

非管理対象インタフェースのポーリング設定との同期, 14

必要なソフトウェア, 5

複数のポーラー, 16

プロパティインポート、ポーリングフラグの設定, 14

分散環境でのインストール, 6

ベンダー固有の OVPI タイプ検出ファイル, 16

ポーラー、複数, 16

ポーリンググループ

説明, 19

デフォルト, 20

ルールベース, 20

ポーリング対象オブジェクト、一覧, 23

ポーリングフラグ, 12

NNM でのポーリング設定との同期, 14

プロパティインポートによる設定, 14

ポーリングポリシーマネージャによる設定, 13

ポーリングポリシー、標準, 21

ポーリングポリシーマネージャ

新規ポーラーの作成, 17

ポーリングフラグの設定, 13

ポーリングルール

設定, 11

説明, 10

ま

マニュアル、一覧, 3

ら

ルールベースのポーリンググループ, 20