

Interface Reporting ifEntry Datapipe

ソフトウェアのバージョン : 2.4

HP OpenView Performance Insight

ユーザーガイド

2007 年 4 月



法律上の表示

保証

Hewlett-Packard 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。Hewlett-Packard 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。

Hewlett-Packard 製品に適用される特定の保証条件の写しは、代理店または営業所で入手できます。

権利の制限

米国政府による利用、複製、情報開示は、DFARS 252.227-7013 「Rights in Technical Data and Computer Software」の条項 (c)(1)(ii) 号で定められた制限の対象となります。

Hewlett-Packard Company

米国

米国国防総省 (DOD) 以外の合衆国政府機関の権利については、FAR 52.227-19(c)(1,2) で規定されています。

著作権

© Copyright 2003 - 2007 Hewlett-Packard Development Company, L.P.

本書のいかなる部分も、Hewlett-Packard Company から事前に書面による許諾を得ることなく、無断で複写、複製、翻訳をしてはならない。マテリアル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。

商標

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。

その他の製品名は、それぞれの商標またはサービスマーク所有者の財産であり、これによって承認されています。

サポート

次の HP OpenView の Web サイトを参照してください。

<http://www.managementsoftware.hp.com/>

これらの Web 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。

HP OpenView のオンラインソフトウェアサポートでは、お客さまが自己解決できるための方法をご提供しています。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 サポート契約を調べる
- 利用可能なサービスに関する情報を確認する
- 他のソフトウェア利用者とディスカッションする
- 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。多くの場合、サポート契約も必要です。

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://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。

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

目次

1	概要	3
	機能拡張および不具合の修正	3
	追加情報の入手先	4
2	データパイプのインストール	5
	円滑なインストールのためのガイドライン	5
	ソフトウェアの前提条件	5
	分散環境	6
	Interface Reporting ifEntry Datapipe のインストール	6
	パッケージの削除	7
3	データパイプの設定	9
	設定タスク	9
	ポーリングルールの設定	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	13
	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	13
	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	14
	NNM との同期化によるポーリングフラグの設定	14
	カスタム SQL スクリプトの作成	15
	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
4	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	20
	IR_map_p	21
5	ポーリング対象オブジェクト	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

1 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。

機能拡張および不具合の修正

次の表に、最近行われた拡張機能と修正点の概要を示します。

バージョン	リリース日	機能拡張および修正点
1.1	2003 年 6 月	Sybase のサポート
2.0	2004 年 4 月	Oracle のサポートを追加
2.1	2004 年 11 月	インタフェースの同期化の拡張機能 (OVPI および NNM)、Sybase および 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
2.2	2005 年 6 月	修正点
2.3	2006 年 5 月	修正点
2.4	2007 年 4 月	パーティションの保守 次のファイルに対する Sybase 15 ROW_COUNT の更新： • install_IR_map_p.sql 新しいアップグレードパッケージ： 修正点 • QXCR1000365606: PropertyData インポートを使用したインタフェースポーリングフラグのリセットができない • QXCR1000219237: IR_map_p : US 以外のロケールの Oracle で失敗する

バージョン	リリース日	機能拡張および修正点
2.4	2007年4月	<ul style="list-style-type: none"> • QXCR1000288184: IR_MAP_P が ORA-06502 を生成: PL/SQL: 数または値エラー • QXCR1000348557: 10G 速度のインタフェースのデバイスに対し正しい情報が収集されない • QXCR1000368656: IR_map_p のエラー率の誤り • QXCR1000368647: 廃棄割合を NULL で計算 • QXCR1000385624: IR_map_p におけるゼロ除算 • QXCR1000351536: IRifEntry_Datapipe のインストール後、IR レポートがデータを表示しない • QXCR1000368656: IR_map_p のエラー率の誤り

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 2.4 リリースノート』
- 『Interface Discovery Datapipe 2.4 ユーザーガイド』
- 『Interface Reporting Report Pack 5.2 ユーザーガイド』
- 『Frame Relay CPE Datapipe 4.2 ユーザーガイド』
- OVPI Report Packs、リリースノート、2007年4月

中心となる製品である OVPI のマニュアルと、OVPI 上で動作するレポート機能および共有パッケージ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://ovweb.external.hp.com/lpe/doc_serv/

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。レポートパックおよびデータパイプのユーザーガイドは、[Performance Insight Reporting Solutions] の下に一覧表示されます。マニュアルのエントリーは、そのマニュアルが Web に掲載された年および月を示します。マニュアルが改訂されて再度掲載されると、日付が変わります。改訂されたマニュアルはその都度掲載されるので、PDF ファイルと Web 版を比較して、新しいバージョンが掲載されている場合はダウンロードしてください。

2 データパイプのインストール

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データパイプのインストール
- データパイプのアンインストール

円滑なインストールのためのガイドライン

レポートパック CD には、最新のレポートパックとデータパイプが含まれています。CD を CD-ROM ドライブに挿入してパッケージ抽出プログラムを起動すると、CD のインストールスクリプトは CD からすべてのパッケージを抽出し、それをシステムの Packages ディレクトリにコピーします。抽出が終わると、インストールスクリプトは、OVPI を起動してパッケージマネージャを開始するように促します。

抽出処理がすでに実行されている場合、Interface Reporting ifEntry Datapipe および「to 2.4」アップグレードパッケージなど、CD 内にあるすべてのパッケージがシステムの Packages ディレクトリ内にすでに保存され、インストールを待機した状態になっています。Interface Reporting ifEntry Datapipe をインストールするには、パッケージマネージャを開始し、画面の指示に従います。



このデータパイプの古いバージョンを現在使用している場合は、アップグレードパッケージをインストールしてバージョン 2.4 にアップグレードします。

ソフトウェアの前提条件

Interface Reporting ifEntry Datapipe 2.4 には、以下の前提条件があります。

- OVPI 5.2
- OVPI 5.2 に提供されているすべてのサービスパック
- Interface Discovery Datapipe 2.4

サービスパックは次の Web サイトからダウンロードできます。

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

Interface Discover Datapipe を先にインストールした後、パッケージマネージャを再起動して Interface Reporting ifEntry Datapipe を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。

Interface Reporting ifEntry Datapipe のインストール

レポートパックの CD を挿入し、指示に従ってパッケージを抽出します。Windows の場合、[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。

抽出が終わると、インストールスクリプトは、OVPI を起動してパッケージマネージャを開始するように促します。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の場合は、この手順を **手順 4** から始めてください。そのほかの場合は、**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 システムの場合は 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]を選択します。

UNIX: root で次のいずれかを入力します。

```
HP-UX: sh /sbin/init.d/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] を選択します。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。[OVPI レポートの配布解除] 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。[OVPI パッケージの場所] ウィンドウが開きます。
- 8 アップグレードする場合は、以下の項目のチェックボックスをクリックします。

UPGRADING_Interface_Reporting_ifEntry_Datapipe_to_2.4

- 9 このデータパイプを初めてインストールする場合は、以下の項目のチェックボックスをクリックします。

Interface_Reporting_ifEntry_Datapipe_2.4

▶ IR_ifEntry_Datapipe には、次のサブパッケージが含まれています。

- IRifEntry_Sync_Interfaces

OVPI および NNM を統合する場合は、このパッケージを必ずインストールする必要があります。

10 [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。

11 [次へ] をクリックします。[概要] ウィンドウが開きます。

12 [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。

13 [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、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。**OVPI Timer** を再起動する前に、再割り当てを行ってください。適切なインタフェース識別子 **index-by** グループへのデバイスの再割り当てについては、『**Interface Discovery Datapipe ユーザーガイド**』を参照してください。

14 必要があれば、デバイスを適切なインタフェース識別子グループに再割り当てします。

15 **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] を選択します。

UNIX: root で次のいずれかを入力します。

```
HP-UX: sh /sbin/init.d/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 システムの場合は **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] を選択します。

UNIX: root で次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 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ます。パッケージの選択を求めるプロンプトが表示されたら、**Interface Reporting ifEntry 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。

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

UNIX: root で次のいずれかを入力します。

HP-UX: sh /sbin/init.d/ovpi_timer start

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

3 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** 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 **Interface Reporting ifEntry Datapipe** 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	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	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。

ポーリングルールを設定を行うには、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。

[Set Polling Rule] 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。

- 5 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。

- 6 [E] を選択して [Enter] を押し、前のメニューに戻ります。

[Configure Polling] メニューが開きます。

- 7 [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。

- 8 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 Reporting 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

[一般的なタスク]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 5.2 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 での「非管理対象インタフェース」のポーリング設定との同期
- 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 **[Apply]** 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せん。

- 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ト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

「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「非管理対象インタフェース」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。**IRifEntry Sync Interfaces** パッケージおよび NNM/OVPI 統合ウィザードの使用の詳細は、『**NNM/Performance Insight Integration Module ユーザーガイド**』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 NNM サーバーで、**install.ovpl** スクリプトを実行して、統合モジュールを設定します。OVPI サーバーの完全修飾名および OVPI Web サーバーのポート番号の入力が求められます。
- 2 OVPI サーバーで、**IRifEntry Sync Interfaces** パッケージをインストールします。第 2 章「データパイプのインストール」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、以下の代替識別子が使用できます。

- **ifDescr** — **index-by-ifDescr** ポーリンググループでサポート
- **ifName** — **index-by-ifName** ポーリンググループでサポート
- **ifAlias** — **index-by-ifAlias** ポーリンググループでサポート

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.txt
```

- 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 がいないもの）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 [編集]>[ポーリンググループ]の順に選択します。表示されたダイアログの上のペインで[同じビュー内にあるすべてのノード]を選択して、下のペインで poller1 を選択します。[Edit] ボタンをクリックします。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 [編集]>[ポーリンググループ]の順に選択します。表示されたダイアログの上のペインで[同じビュー内にあるすべてのノード]を選択して、下のペインで poller2 を選択します。[Edit] ボタンをクリックします。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。

4 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
GENMIB2IF_Datapipe	MIB-2 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。
index-by-ifAlias	GENMIB2IF_Datapipe グループのサブセット。識別子として ifAlias を使用するすべてのノードを含みます。
index-by-ifDescr	GENMIB2IF_Datapipe グループのサブセット。識別子として ifDescr を使用するすべてのノードを含みます。
index-by-ifIndex	GENMIB2IF_Datapipe グループのサブセット。 index-by-ifAlias、index-by-ifDescr、または index-by-ifName 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 ifIndex です。このグループの編集は必要ありません。
index-by-ifName	GENMIB2IF_Datapipe グループのサブセット。識別子として ifName 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.
```


5 ポーリング対象オブジェクト

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 (下記)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 でソートしたものです。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の値。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。
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。

MIB オブジェクト名	オブジェクト ID	説明
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、1(動作)、2(停止)、3(テスト)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、1(動作)、2(停止)、3(テスト)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オクテットの総数(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)。
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)。

MIB オブジェクト名	オブジェクト ID	説明
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。</p> <p>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オクテットの総数(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。

索引

C

Cisco Catalyst スイッチ、代替識別子, 15
Cisco ルーター、代替識別子, 15

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, 15

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, 24
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, 23
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
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 スタアドプロシージャ, 21
IRifEntry_Sync_Interfaces, 6
IRifEntry Sync Interfaces, 14, 15

K

K_IFEntry_Disc プロパティテーブル, 21

O

OVPI Timer, 7

P

poll_flag_ifEntry, 21
ProCurve スイッチ、代替識別子, 15

S

SQL スクリプト、ポーリングフラグの設定用, 15

あ行

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

か行

拡張 MIB-2 テーブル, オブジェクト, 26

さ行

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, 4

設定

複数のポーラー, 16

ポーリングルール, 10

た行

代替識別子, 15

代替識別子、変更, 15

タイプ検出ファイル, 15

データパイプ

インストール, 6

削除, 7

データパイプのアンインストール, 7

な行

ネットワークノードマネージャ (NNM)

OVPI との統合, 6

ポーリングフラグの設定, 14

は行

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

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, 15

ポーラー、複数, 16

ポーリンググループ

説明, 19

デフォルト, 19

ルールベース, 20

ポーリングフラグ, 13

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, 20

ポーリングポリシーマネージャ

新規ポーラーの作成, 16

ポーリングフラグの設定, 13

ポーリングルール

設定, 10

説明, 10

ま行

マニュアル、一覧, 4

ら行

ルールベースのポーリンググループ, 20