Interface Reporting ifEntry Datapipe

Software Version: 2.5

HP Performance Insight

User Guide

Legal Notices

Warranty

Restricted Rights Legend

Copyright Notices

Trademark Notices

Support

Please visit the HP Software web site at:

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

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

Online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest

- Submit and track progress on support cases

- Manage a support contract

- Look up HP support contacts

- Review information about available services

- Enter discussions with other software customers

- Research and register for software training

Most of the support areas require that you register as an HP Passport user and log in. Many also require a support contract.

To find more information about access levels, go to:

http://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

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

Contents

1 Overview

The Interface Reporting ifEntry Datapipe collects MIB-II ifEntry and ifxEntry data from SNMP manageable devices. In addition to collecting data, the IR ifEntry Datapipe performs the following functions:

- Supports interface re-indexing detected by the Interface Discovery Datapipe

- Performs directed-instance polling

- When values exceed 32 bits, supports high-capacity counters

- Supports interface synchronization with NNM when NNM is configured to poll selected interfaces per device

The IR ifEntry Datapipe does not require configuration. The configuration options described in Chapter 3 are optional.

Enhancements and Defect Fixes

The following table outlines recent enhancements and defect fixes.

Version	Release Date	Enhancements/Defect Fixes
1.1	June 2003	Sybase support
2.0	April 2004	Added Oracle support
2.1	November 2004	Interface synchronization enhancements (OVPI and NNM), Sybase and Oracle; minor fixes to formulas that calculate discard, error, and utilization percentages
2.2	June 2005	Defect fixes
2.3	May 2006	Defect fixes
2.4	April 2007	Partition maintenance Sybase 15 ROW_COUNT updates to this file: • install_IR_map_p.sql New upgrade package: • UPGRADE_IRifEntry_Datapipe_to_2.4 Defect fixes: • QXCR1000365606: Unable to reset Interface Polling Flags via PropertyData import • QXCR1000219237: IR_map_p: Fails on non-US locale Oracle

Version	Release Date	Enhancements/Defect Fixes
2.4	April 2007	• QXCR1000288184: IR_MAP_P produces ORA-06502: PL/SQL: numeric or value error • QXCR1000348557: Do not collect correct info for devices with 10G speed interfaces • QXCR1000368656: Error percentage wrong in IR_map_p • QXCR1000368647: Calculating discards percent with NULLs • QXCR1000385624: Divide by zero in IR_map_p • QXCR1000351536:IR reports show no data after installing IRifEntry_Datapipe • QXCR1000368656: Error percentage wrong in IR_map_p
2.5	October 2007	New upgrade package: • UPGRADE_IRifEntry_Datapipe_to_2.5 Defect fixes: • QXCR1000431418: IRifEntry Datapipe should handle devices in more than one index-by group • QXCR1000415791: Average utilization higher than peak on Interface NRT report • QXCR1000407857: IR_map_p should use 64-bit counters for Packets, Discards and Errors • QXCR1000348557: Do not collect correct info for devices with 10G speed interfaces

Sources for Additional Information

The following documents are related to this manual:

- *Interface Reporting ifEntry Datapipe 2.5 Release Notes*

- *Interface Discovery Datapipe 2.5 User Guide*

- *Interface Reporting Report Pack 5.3 User Guide*

- *Frame Relay CPE Datapipe 4.3 User Guide*

- *PI Report Packs, CD-ROM Release Notes, October 2007*

Manuals for the core product, OVPI, and manuals for the reporting solutions and shared packages that run on OVPI, can be downloaded from this site:

http://ovweb.external.hp.com/lpe/doc_serv/

The user guides for the core product are listed under **Performance Insight**. The user guides for report packs and datapipes are listed under **Performance Insight Report Packs**. The entry for a manual indicates the month and year it was posted to the web. If a manual is revised and reposted, the date will change. Since revised manuals are reposted from time to time, be sure to compare your PDF to the web edition and download the web edition if it is newer.

2 Installing the Datapipe

This chapter covers the following topics:

* Guidelines for a smooth installation
* Using Package Manager to install the IR ifEntry Datapipe
* Uninstalling the datapipe

Guidelines for a Smooth Install

The report pack CD contains the latest report packs and datapipes. When you insert the CD in the CD-ROM drive and launch the package extraction program, the install script on the CD extracts every package from the CD and copies the results to the Packages directory on your system. When the extract finishes, the install script prompts you to launch OVPI and start Package Manager.

If the extraction step has already taken place, the every package on the CD, including the IR ifEntry Datapipe, the "to 2.5" upgrade package, and IRifEntry Sync Interfaces, is now on your system in the Packages directory. To install these packages, start Package Manager and follow the on-screen instructions. If you are currently running an earlier version of the IR ifEntry Datapipe, simply upgrade to version 2.5 by installing the upgrade package.

Software Prerequisites

The IR ifEntry Datapipe 2.5 has the following prerequisites:

* Performance Insight 5.3
* All available service packs for PI 5.3
* Interface Discovery Datapipe 2.5

Service packs are available at the following website:

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

Distributed Systems

If you plan to install the Interface Reporting Report Pack on multiple servers in a distributed environment, datapipe installation is more complicated. Make sure that you install the IR ifEntry Datapipe on every OVPI server that performs polling. If the central server is not polling, there is no need to install the IR ifEntry Datapipe on the central server.

Installing the IR ifEntry Datapipe

Insert the report pack CD and follow the instructions for extracting packages. On Windows, the instructions appear in a Main Menu that opens automatically; on UNIX, you must log in as root, mount the CD (if the CD does not mount automatically), navigate to the top level of the CD directory, and then run the `./setup` command.

After the extract finishes, the install script prompts you to launch OVPI and start Package Manager. If you just extracted packages from the CD and the Package Manager install wizard is running, start this procedure at step 4. Otherwise, begin at step 1.

1   Log in to the system. On UNIX systems, log in as root.

2   Stop OVPI Timer and wait for processes to terminate.

   Windows: Select **Settings > Control Panel** > **Administrative Tools > Services**.

   UNIX: As root, type one of the following:

   HP-UX: `sh /sbin/init.d/ovpi_timer stop`

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

3   From the Management Console, select **Tools > Package Manager**. The Package Manager welcome window opens.

4   Click **Next.** The Package Location window opens.

5   Click **Install**. Approve the default destination directory or browse to a different directory if necessary.

6   Click **Next.** The Report Deployment window opens. Type your OVPI username and password,

7   Click **Next.** The Package Selection window opens.

8   If you are upgrading, click the check box for:

   UPGRADE_IRifEntry_Datapipe_to_ 2.5

   If you are installing this datapipe for the first time, click the check box for:

   IRifEntry_Datapipe_ 2.5

   ▶   The IR_ifEntry Datapipe includes the following sub-package:

      IRifEntry_Sync_Interfaces

      The IRifEntry Sync Interfaces package is designed to upload a list of managed and unmanaged interfaces from NNM, set the polling flag for managed interfaces, and clear the flag for unmanaged interfaces. Installing this package is mandatory only when:
      * PI and NNM are integrated
      * NNM is configured to poll selected interfaces per device

9   Click **Next.** The Type Discovery window opens. Clear the default.

   ▶   If you are also installing the Interface Discovery Datapipe, refer to the *Interface Discovery Datapipe User Guide* for information about the Type Discovery option.

10 Click **Next.** The Selection Summary window opens.

11 Click **Install.** The Installation Progress window opens and the install begins. When the install finishes, a package installation complete message appears.

12 Click **Done** to return to the Management Console..

> ▶ If you are installing the Interface Discovery Datapipe, you may want to re-assign devices to the appropriate interface identifier index-by group. Do the re-assignment before restarting OVPI Timer. For details about re-assigning devices to the appropriate interface identifier index-by group, see the *Interface Discovery Datapipe User Guide*.

13 If required, reassign devices to the appropriate interface identifier groups.

14 Restart OVPI Timer.

Windows: Select **Settings > Control Panel** > **Administrative Tools > Services**.

UNIX: As root, type one of the following:

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

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

The IR ifEntry Datapipe is now installed on your system.

Package Removal

Follow these steps to uninstall the Interface Reporting ifEntry Datapipe.

1 Log in to the system. On UNIX systems, log in as root.

2 Stop OVPI Timer and wait for processes to terminate.

Windows: Select **Settings > Control Panel** > **Administrative Tools > Services**.

UNIX: As root, type one of the following:

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

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

3 Start Package Manager. The Package Manager install wizard opens.

4 Follow the on-screen directions for package removal; when prompted to select a package, select the IR ifEntry Datapipe. When the uninstall finishes, a package removal complete message appears.

5 Click **Done** to return to the Management Console.

6 Restart OVPI Timer.

Windows: Select **Settings > Control Panel** > **Administrative Tools > Services**.

UNIX: As root, type one of the following:

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

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

3 Configuring the Datapipe

This chapter covers the following topics:

- Configuration tasks and how to accomplish them
- Configuring a polling rule
- Accessing forms bundled with Interface Reporting
- Exporting and importing property data
- Setting the poll flag
- Changing the interface identifier
- Configuring the IR ifEntry Datapipe for multiple pollers

Configuration Tasks

The following table maps each configuration task to one or more ways to accomplish the task.

Task	Means to Accomplish
Configure a polling rule	Configuration Tool
Set the data filtering mode	Administer Interface Reporting (form)
Set the provisioning mode	Administer Interface Reporting (form)
Set the message level	Administer Interface Reporting (form)
Export and import property data	IR Configuration Tool
Set the polling flag	1 Administer Interface Poll Flag (form) 2 Polling Policy Manager 3 Property import for the IR Report Pack 4 Synchronize polling with NNM 5 Custom SQL script
Disable protocol monitoring	Change Protocol Defaults (form)
Change the interface identifier	1 Interface Discovery Datapipe 2 Polling Policy Manager
Configure a datapipe for multiple pollers	1 OVPI group_manager 2 OVPI collection_manager 3 Polling Policy Manager

Configuring a Polling Rule

Polling is conducted in accordance with one of the following five rules:

(1) Poll All Active Interfaces

Poll all interfaces where the ifAdminStatus is set to Active.

(2) Poll All Interfaces

Poll all interfaces, regardless of ifAdminStatus or protocol. (This is the default rule when the Interface Reporting ifEntry Datapipe is installed.)

(3) Poll All Active Interfaces with Monitored Protocols

Default. Poll all interfaces where the ifAdminStatus is set to Active and the protocol (ifType) is configured for monitoring in the KIR_protocol_config table. You can disable monitoring on protocols that do not require monitoring.

➤ The default is Poll All Active Interfaces with Monitored Protocols.

(4) Poll All Interfaces with Monitored Protocols

Poll all interfaces where the protocol (ifType) is configured for monitoring in the KIR_protocol_config table. You can disable monitoring on protocols that do not require monitoring.

(5) Poll Flagged Interfaces Only

Poll a subset of interfaces, flagged interfaces only. This setting allows individual interfaces to be provisioned for collection.

Follow these steps to configure one of the preceding polling rules:

1 Log in to the system. On UNIX, log in as trendadm.

2 Start the IR Configuration Tool from the command line:

   UNIX:

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

   Windows:

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

   The IR Configuration Tool main menu opens.

3 Select option 1, Configure Polling, and press **Enter**.

   The Configure Polling menu opens.

4 Select option 1, Set Polling Rule, and press **Enter**.

   The Set Polling Rule menu opens, displaying the current polling rule and the date and time the rule was last changed.

5 Select the option you want and press **Enter**.

   As database views that implement the polling rule are rebuilt, messages appear; then the Set Polling Rule menu opens again, showing the new polling rule.

6   Select **E** and press **Enter** to return to the previous menu.

      The Configure Polling menu opens.

7   Select **E** and press **Enter** to return to the previous menu.

      The IRCT main menu opens.

8   Select an option or select **E** and press **Enter** to exit.

►   If you apply the "Poll Flagged Interfaces Only" rule, you must also set the polling flag for the interfaces you want to poll.

Accessing Forms Bundled with Interface Reporting

Follow these steps to access the forms bundled with Interface Reporting:

1   Start the Management Console.

2   Click the **Objects** icon. The Object/Property Management window opens.

3   Select a device.

      The following forms are listed under **Object Specific Tasks**:

      •   Change Protocol Defaults

         ►   By default, all protocols (ifTypes) are monitored, which means that data will be collected for all protocols. If you are not interested in certain protocols, use the Change Protocol Defaults form to disable collection and reporting.

      •   Change Interface

      •   Change Interface Customer

      •   Change Interface Description

      •   Administer Interface Poll Flag

      The following form is listed under **General Tasks**:

      •   Administer Interface Reporting

For more information about forms, see Chapter 6, Using Change Forms, in the *Interface Reporting Report Pack 5.3 User Guide*.

Exporting and Importing Property Data

The IR Configuration Tool makes it easy for you to export existing property data from the Interface Reporting Report Pack and import new property data into the Interface Reporting Report Pack. When you use the IR Configuration Tool to export or import property data, the data file will be exported to, and imported from, the default directory for property data owned by the Interface Reporting Report Pack, which is:

     `$DPIPE_HOME/data/PropertyData`

Follow these steps to export or import property data:

1 Log in to the system. On UNIX, log in as trendadm.

2 Start the IR Configuration Tool from the command line:

   UNIX:

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

   Windows:

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

   The IRCT main menu opens.

3 Select option 2, Export or Import Interface Reporting Property Data, and press **Enter**.

   The Property Export and Import menu opens.

4 Do one of the following:

   Select **1. Export Interface Reporting Property Data** to export.

   Select **2. Import Interface Reporting Property Data** to import.

5 When finished, select **E** and press **Enter** to return to the previous menu.

   The IR Configuration Tool main menu opens.

6 Select an option, or select **E** and press **Enter** to exit.

Setting the Polling Flag

To set the polling flag, use any of the following methods:

- Administer Interface Poll Flag (a form)
- Polling Policy Manager
- Property import for the Interface Reporting Report Pack
- Synchronize with the "unmanaged interfaces" polling configuration on NNM
- Create a custom SQL script

Using Polling Policy Manager to Set the Polling Flag

When the Poll Flagged Interfaces Only rule is enabled, the IR ifEntry Datapipe collects statistics only from interfaces that have a polling flag set in the main interface property table, K_IFEntry_Disc.poll_flag_ifEntry. Follow these steps to use Polling Policy Manager to modify the polling flag:

1 Determine which interfaces on which devices will be polled and which interfaces on which devices will not be polled.

2 Log in to the system. On UNIX, log in as root.

3 Start Polling Policy Manager.

4 Select **Edit > Descriptions...** The Edit Descriptions window opens.

5 Select the Property Group K_IFEntry_Disc and click **Rule: Edit...** The Create Rule window opens.

6   Select or enter a rule; for example, to return higher speed interfaces only, use interface_speed >= 100000000, and click **OK**.

7   Click **View Resulting Instances**. The instances will appear in the lower pane.

8   From the returned list, select all the interfaces that are to be polled and click **Edit Instance Values...** The Edit Instance window opens.

9   Select the Column poll_flag_ifEntry, enter a value of 1, and click **Apply;** then click **Close** to return to the Edit Descriptions window.

10 Click **View Resulting Instances**. The updated instances appear in the lower pane.

11 Click **Close**.

12 If the Poll Flagged Interfaces Only rule is not already enabled, enable it now.

Using Property Import to Set the Polling Flag

When the Poll Flagged Interfaces Only rule is enabled, the IR ifEntry Datapipe collects statistics only from interfaces that have a polling flag set in the main interface property table. Follow these steps to set the polling flag using the Interface Reporting Property Import:

➤   If Data Filtering mode is enabled, so that only data for provisioned interfaces is collected, the polling flag is set automatically for each interface included in the property import file.

1   Determine which interfaces on which devices will be polled, and which interfaces on which devices will not be polled.

2   Log in to the system. On UNIX, log in as root.

3   Generate a new Interface Reporting property import file with the interfaces that are to be polled, or, if the interfaces have already been discovered, export the property file and edit it to include only the desired interfaces.

4   Use the Administer Interface Reporting form to check whether data filtering mode is on or off. If data filtering mode is not already enabled, enable it now.

5   Import the Interface Reporting property data you produced in step 3.

6   If the Poll Flagged Interfaces Only rule is not already enabled, enable it now.

Set the Polling Flag by Synchronizing with NNM

When the Poll Flagged Interfaces Only rule is enabled, the IR ifEntry Datapipe will collect statistics only from interfaces that have a polling flag set in the main interface property table. This flag can be set by synchronizing with the "unmanaged interfaces" polling configuration on NNM. NNM allows you to define unmanaged interfaces two ways, through the user interface, and by using a special add-on, rules-based tool designed for this purpose.

The IRifEntry Sync Interfaces package is designed to upload the managed and unmanaged interfaces list from NNM, set the polling flag for managed interfaces, and clear the flag for unmanaged interfaces. When you install the IRifEntry Sync Interfaces package, these events take place:

•   A complete list of managed and unmanaged interfaces is downloaded from NNM.

•   The interface polling flag is set (based on the NNM configuration).

•   OVPI Timer entries are configured for:

— A daily synchronization of all interfaces that have undergone a change in management status during the previous 24 hours

— A full re-synchronization once a month

Follow these steps to set the polling flag by synchronizing with NNM:

1   On the NNM server, run the install.ovpl script. You will be prompted to enter the fully-qualified name of the PI server and the port number of the PI web server.

2   On the PI server, install the IRifEntry Sync Interfaces package. Follow the procedure provided in Chapter 2, Installing the Datapipe.

3   If the Poll Flagged Interfaces Only rule is not already enabled, enable it now.

For details about the the NNM/PI integration wizard, refer to the *NNM/Performance Insight Integration User Guide*.

Creating a Custom SQL script

If polling configuration is held in a provisioning or configuration management system, develop a custom SQL script to set the poll flag (K_IFEntry_Disc.poll_flag_ifEntry).

Changing the Interface Identifier (Re-Indexing Support)

The Interface Discovery Datapipe provides the following alternate identifiers:

* ifDescr—supported by the index-by-ifDescr polling group

* ifName—supported by the index-by-ifName polling group

* ifAlias—supported by the index-by-ifAlias polling group

To enable re-indexing, select an alternate identifier (just *one* alternate identifier) that is unique for the device and persistent.

➤   Do **not** assign a device to more than one group. Only one alternate identifier per device is allowed.

The Interface Discovery Datapipe includes vendor-specific OVPI Type Discovery files that automatically assign certain types of devices to an alternate interface identifier, so some of the selection work is already done for you. The following table contains a list of device types and pre-assigned identifiers.

Device Type	Identifier Assigned by OVPI Type Discovery
Cisco Catalyst switches	ifName
Cisco routers	ifDescr
HP ProCurve switches (except 800T)	ifName

The index-by-ifIndex group is the default group. This group contains all the devices in the GENMIB2IF_Datapipe group that are not in the index-by-ifAlias, index-by-ifDescr, or index-by-ifName groups.

Follow these steps to change an interface identifier:

1   For each device type, choose the interface identifier.

2   Assign each device of that type to the appropriate index-by group.

    a   Start Polling Policy Manager.

    b   Select **Edit** > **Polling Groups**...

    c   Select **All Nodes of the Same Type** in the top pane and the required index-by group in the bottom pane.

    d   Click **Edit**.

    e   Select the nodes to be assigned to this group in the left-hand pane, and click **>>** to move them into the group. Click **OK** when complete.

3   Repeat the process above for other device types and index-by groups as necessary.

If a device is assigned to an index-by group, but the identifiers are not unique, a duplicate identifier message, similar to the following message, will be inserted in the trendlog.

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

Configuring the Datapipe for Multiple Pollers

The IR ifEntry Datapipe requires additional configuration for use with multiple pollers. The standard installation configures four groups and four polling policies for a single poller. For use with multiple pollers, additional sets of groups and polling policies must be configured for each additional poller.

The datapipe package includes additional group and collection policy files for two pollers. If you need to configure the datapipe for more than two pollers against a single database, follow these steps to generate additional sets of group files and policies:

1   Log in as the OVPI Administrator (trendadm).

2   Navigate to the IR ifEntry Datapipe package directory:

3   Run the following command:

    trend_proc -f create_groups_multi_pollers.pro

4   Run the following command:

collection_manager -import -file IRifEntry_Collection_Policies_multi_pollers.txt

5   Start Polling Policy Manager and select **Edit > Datapipe Installations...**

6   Click **Create...** to create new pollers.

7   Add the hostname and (optionally) the IP address of each poller and click **OK**.

8   When all pollers have been added, click **Close** to exit the Edit Datapipe Installations window.

9   Click **All** in the left panel to display all polling policies.

10   Delete the original if*_Collection policies (those without a _poller1 or _poller2 at the end of the name).

11 Modify the new if*_Collection_poller1 policies by double-clicking the policy.

Change the **Polling Assigned to** setting to be the hostname of poller1.

12 Modify the new if*_Collection_poller2 policies by double-clicking the policy.

Change the **Polling Assigned to** setting to be the hostname of poller2.

At this point, the 15-minute policies should look similar to those in the following table.

Policy Name	Table Alias Name for Data to be Collected	Group Name to Collect From	Poll From
ifAlias_Collection_poller1	xV_IRifEntry_ifAlias	ifAlias-poller1	Hostname of poller1
ifDescr_Collection_poller1	xV_IRifEntry_ifDescr	ifDescr-poller1	Hostname of poller1
ifIndex_Collection_poller1	xV_IRifEntry_ifIndex	ifIndex-poller1	Hostname of poller1
ifName_Collection_poller1	xV_IRifEntry_ifName	ifName-poller1	Hostname of poller1
ifAlias_Collection_poller2	xV_IRifEntry_ifAlias	ifAlias-poller2	Hostname of poller2
ifDescr_Collection_poller2	xV_IRifEntry_ifDescr	ifDescr-poller2	Hostname of poller2
ifIndex_Collection_poller2	xV_IRifEntry_ifIndex	ifIndex-poller2	Hostname of poller2
ifName_Collection_poller2	xV_IRifEntry_ifName	ifName-poller2	Hostname of poller2

13 Select **Edit > Polling Groups... > All Nodes in Same View** in the upper pane, then select poller1 in the lower pane. Assign nodes to be collected by poller1 to the poller1 view.

14 Select **Edit > Polling Groups... > All Nodes in Same View** in the upper pane, then select poller2 in the lower pane. Assign nodes to be collected by poller2 to the poller2 view.

4 Groups and Policies

This chapter covers the following topics:

- Polling groups
- Rules-based polling groups
- Polling policies
- The IR_map_p stored procedure

Polling Groups

The IR ifEntry Datapipe uses the same OVPI Polling "Type" Groups that the Interface Discovery Datapipe uses. Polling groups make re-indexing support possible. They are created when the Interface Discovery Datapipe is installed.

Polling Group	Description
GENMIB2IF_Datapipe	Top-level group for all nodes that support MIB-II.
index-by-ifAlias	Subset of the GENMIB2IF_Datapipe group; contains all nodes that should use ifAlias as the identifier.
index-by-ifDescr	Subset of the GENMIB2IF_Datapipe group; contains all nodes that should use ifDescr as the identifier.
index-by-ifIndex	Subset of the GENMIB2IF_Datapipe group; contains all the devices that do not belong to the index-by-ifAlias, index-by-ifDescr, or index-by-ifName groups. Unless a device is assigned to a different group, it will default to ifIndex as the interface identifier. Editing this group is not necessary.
index-by-ifName	Subset of the GENMIB2IF_Datapipe group; contains all nodes that should use ifName as the identifier.

The index-by-ifIndex group is the default. This group does not control polling; rather, it shows you which devices will use ifIndex as the interface identifier. Since this group is automatically repopulated by the system once per hour, any device that is re-assigned to a different index-by group may not immediately disappear from index-by-ifIndex. However, since the data collection for index-by-ifIndex is based on devices in the GENMIB2IF_Datapipe group, the device will be polled only from its new index-by group.

Rules-Based Polling Groups

In addition to the five basic polling groups, the IR ifEntry Datapipe adds four groups that are derived from the basic groups but also contain rules for rules-based polling.

Group Name	Database View	Description
ifAlias-all	KV_IFEntry_by_ifAlias	Derived from the index-by-ifAlias group with polling rules applied.
ifDescr-all	KV_IFEntry_by_ifDescr	Derived from the index-by-ifDescr group with polling rules applied.
ifIndex-all	KV_IFEntry_by_ifIndex	Derived from the devices in the GENMIB2IF_Datapipe group that are not in index-by-ifAlias, index-by-ifDescr, or index-by-ifName, with polling rules applied.
ifName-all	KV_IFEntry_by_ifName	Derived from the index-by-ifName group with polling rules applied.

To see the list of interfaces that will be polled in each group, use SQL to select from the database views in the table above.

Polling Policies

A standard installation of the IR ifEntry Datapipe enables a set of four 15-minute polling policies, one for each index-by group. Changing these policies is not necessary, unless you need to enable support for multiple pollers.

Policy Name	Table Alias Name for Data to be Collected	Group Name to Collect From
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

The IR ifEntry Datapipe adds a flag column called poll_flag_ifEntry to the Interface Discovery Datapipe property table K_IFEntry_Disc. Provisioning this flag will control directed-instance polling.

IR_map_p

The IR ifEntry Datapipe uses a stored procedure called IR_map_p to transfer collected data to the Interface Reporting Report Pack. IR_map_p is called at the end of the raw-to-rate conversion. IR_map_p and the conversion procedure run at the end of each of the four polling policies above. While they are running, these procedures lock tables to ensure that no data is lost. As a result, it is not unusual for one or more instances of the raw-to-rate stored procedure to generate lock errors indicating that an instance is already running. Here is an example of the error message:

```
"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 also generates standard information messages with the number of rows transferred to the Interface Reporting Report Pack. For example:

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

5 Polled Objects

The following tables provide the name, identifier, and description of polled objects. Table 1 (below) provides a list of objects, sorted by object ID, in the ifEntry MIB table. Table 2 (page 24) provides a list of objects, organized by object ID, in the MIB-II Extensions table, also known as the ifXTable.

Table 1: Objects in ifEntry

MIB Object Name	OID	Description
ifIndex	1.3.6.1.2.1.2.2.1.1	A unique value for each interface, ranging between 1 and the value of ifNumber. The value for each interface must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.
ifDescr	1.3.6.1.2.1.2.2.1.2	A text string containing information about the interface. This string should include the name of the manufacturer, the product name, and the version of the hardware interface.
ifType	1.3.6.1.2.1.2.2.1.3	The type of interface, distinguished according to the physical/link protocol(s) immediately below the network layer in the protocol stack.
ifMtu	1.3.6.1.2.1.2.2.1.4	The size of the largest datagram that can be sent/received on the interface, specified in octets. For interfaces that are used for transmitting network datagrams, this is the size of the largest network datagram that can be sent on the interface.
ifSpeed	1.3.6.1.2.1.2.2.1.5	An estimate of the interface's current bandwidth in bits per second. For interfaces that do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth.
ifPhysAddress	1.3.6.1.2.1.2.2.1.6	The interface's address at the protocol layer immediately below the network layer in the protocol stack. For interfaces that do not have such an address (e.g., a serial line), this object should contain an octet string of zero length.

MIB Object Name	OID	Description
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	The desired state of the interface. In the testing state, no operational packets can be passed. States are 1 (up) 2 (down) 3 (testing)
ifOperStatus	1.3.6.1.2.1.2.2.1.8	The current operational state of the interface. In the testing state, no operational packets can be passed. States are 1 (up) 2 (down) 3 (testing)
ifLastChange	1.3.6.1.2.1.2.2.1.9	The value of sysUpTime at the time the interface entered its current operational state. If the current state was entered before the last re-initialization of the local network management subsystem, this object contains a zero value.
IfInOctets	1.3.6.1.2.1.2.2.1.10	The total number of octets received on the interface, including framing characters.
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	The number of subnetwork-unicast packets delivered to a higher-layer protocol.
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	The number of non-unicast (i.e., subnetwork-broadcast or subnetwork-multicast) packets delivered to a higher-layer protocol.
ifInDiscards	1.3.6.1.2.1.2.2.1.13	The number of inbound packets that were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.
ifInErrors	1.3.6.1.2.1.2.2.1.14	The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.
ifUnknownProtos	1.3.6.1.2.1.2.2.1.15	The number of packets received via the interface that were discarded because of an unknown or unsupported protocol.
ifOutOctets	1.3.6.1.2.1.2.2.1.16	The total number of octets transmitted out of the interface, including framing characters.
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address, including those that were discarded or not sent.
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	The total number of packets that higher-level protocols requested be transmitted to a non-unicast (i.e., a subnetwork-broadcast or subnetwork-multicast) address, including those that were discarded or not sent.

MIB Object Name	OID	Description
ifOutDiscards	1.3.6.1.2.1.2.2.1.19	The number of outbound packets that were chosen to be discarded even though no errors were detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.
ifOutErrors	1.3.6.1.2.1.2.2.1.20	The number of outbound packets that could not be transmitted because of errors.
ifOutQLen	1.3.6.1.2.1.2.2.1.21	The length of the output packet queue (in packets).
ifSpecific	1.3.6.1.2.1.2.2.1.22	A reference to MIB definitions specific to the particular media being used to realize the interface. For example, if the interface is realized by an ethernet, the value of this object refers to a document defining objects specific to ethernet. If this information is not present, its value should be set to the OBJECT IDENTIFIER { 0 0 }, which is a syntactically valid object identifier, and any conforming implementation of ASN.1 and BER must be able to generate and recognize this value.

Table 2: Objects in MIB-II Extensions

MIB Object Name	OID	Description
ifName	1.3.6.1.2.1.31.1.1.1.1	The text name of the interface. The value of this object should be the name of the interface as assigned by the local device and should be suitable for use in commands entered at the device's console. This might be a text name, such as "le0", or a simple port number, such as 1, depending on the interface naming syntax of the device. If several entries in the ifTable together represent a single interface as named by the device, each will have the same value of ifName. If there is no local name, or if this object is otherwise not applicable, this object contains a 0-length string.
ifAlias	1.3.6.1.2.1.31.1.1.1.18	This object is an alias name for the interface as specified by a network manager, and provides a nonvolatile "handle" for the interface. On the first instantiation of an interface, the value of ifAlias associated with that interface is the zero-length string. When a value is written into an instance of ifAlias through a network management set operation, the agent must retain the supplied value in the ifAlias instance associated with the same interface for as long as that interface remains instantiated, even if the network management system reinitializes/reboots. An example of the value that a network manager might store in this object for a WAN interface is the (Telco's) circuit number/identifier of the interface. Some agents may support write-access only for interfaces having particular values of ifType. An agent that supports write access to this object is required to keep the value in non-volatile storage, but it may limit the length of new values, depending on how much storage is already occupied by the current values for other interfaces.
ifHCInOctets	1.3.6.1.2.1.31.1.1.1.6	The total number of octets received on the interface, including framing characters. This object is a 64-bit version of ifInOctets.
ifHCInUcastPkts	1.3.6.1.2.1.31.1.1.1.7	The number of packets, delivered by this sub-layer to a higher (sub-)layer, that were not addressed to a multicast or broadcast address at this sub-layer. This object is a 64-bit version of ifInUcastPkts.

MIB Object Name	OID	Description
ifHCInMulticastPkts	1.3.6.1.2.1.31.1.1.1.8	The number of packets, delivered by this sub-layer to a higher (sub-)layer, that were addressed to a multicast address at this sub-layer. For a MAC layer protocol, this includes both Group and Functional addresses. This object is a 64-bit version of ifInMulticastPkts.
ifHCInBroadcastPkts	1.3.6.1.2.1.31.1.1.1.9	The number of packets, delivered by this sub-layer to a higher (sub-)layer, that were addressed to a broadcast address at this sub-layer. This object is a 64-bit version of ifInBroadcastPkts.
ifHCOutOctets	1.3.6.1.2.1.31.1.1.1.10	The total number of octets transmitted out of the interface, including framing characters. This object is a 64-bit version of ifOutOctets.
ifHCOutUcastPkts	1.3.6.1.2.1.31.1.1.1.11	The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address, including those that were discarded or not sent. This object is a 64-bit version of ifOutUcastPkts.
ifHCOutMulticastPkts	1.3.6.1.2.1.31.1.1.1.12	The total number of packets that higher-level protocols requested be transmitted, and that were addressed to a multicast address at this sub-layer, including those that were discarded or not sent. For a MAC layer protocol, this includes both Group and Functional addresses. This object is a 64-bit version of ifOutMulticastPkts.
ifHCOutBroadcastPkts	1.3.6.1.2.1.31.1.1.1.13	The total number of packets that higher-level protocols requested be transmitted, and that were addressed to a broadcast address at this sub-layer, including those that were discarded or not sent. This object is a 64-bit version of ifOutBroadcastPkts.

Index

K