

HP OpenView Performance Insight

Interface Reporting ifEntry Datapipe 사용자 가이드

소프트웨어 버전 : 2.1

Reporting and Network Solutions 7.0



2004 년 11 월

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

알림

보증

Hewlett-Packard 는 이 문서와 관련하여 상업성이나 특정 목적에의 적합성에 대한 묵시적 보증 등 어떤 종류의 보증도 하지 않습니다. *Hewlett-Packard* 는 이 문서의 오류나 이 자료의 공급, 수행 또는 사용에 따른 직접적, 간접적, 특수, 부수적 또는 파생적 손해에 대해 어떤 종류의 책임도 지지 않습니다. *Hewlett-Packard* 제품에 적용할 수 있는 특별 보증 조건의 복사본은 현지의 판매 및 서비스 센터에서 얻을 수 있습니다.

제한된 권한 설명

미국 정부에 의한 사용, 복제 또는 공개는 DFARS 252.227-7013 의 기술 데이터 및 컴퓨터 소프트웨어 권리 조항 (c) (1) (ii) 에 따라 제한을 받습니다.

Hewlett-Packard Company

United States of America

비 DOD 미국 정부 부서 및 대리처의 권리는 FAR 52.227-19(c)(1,2) 에 따라 제한을 받습니다.

저작권 공고

© Copyright 1993-2004 Hewlett-Packard Development Company, L.P.

Hewlett-Packard Company 의 사전 서면 승인 없이는 이 문서의 어떤 부분도 복사, 복제 또는 다른 언어로 번역할 수 없습니다. 이 문서의 내용은 통지 없이 변경될 수 있습니다.

AirMedia, Inc 의 소프트웨어가 포함되어 있습니다.

© Copyright 1996 AirMedia, Inc

상표 공고

OpenView 는 Hewlett-Packard Development Company, L.P 의 미국 등록 상표입니다.

Java™은 Sun Microsystems, Inc 의 미국 등록 상표입니다.

Oracle® 은 Oracle Corporation, Redwood City, California 의 미국 등록 상표입니다.

UNIX® 는 Open Group 의 등록 상표입니다.

Windows® 및 Windows NT® 는 Microsoft Corporation 의 미국 등록 상표입니다.

지원

다음의 주소로 HP OpenView 웹 사이트를 방문하십시오 .

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

여기서 연계 정보 및 HP OpenView 가 제공하는 제품 및 서비스에 대한 세부사항을 찾을 수 있습니다 . 지원 사이트로 가려면 **지원**을 클릭하십시오 . 지원 사이트는 사용자 업무를 관리하는데 필요한 대화식 기술 지원 도구에 액세스하는 빠르고 효율적인 방법을 제공합니다 . 유가 지원 고객으로서 이 지원 사이트를 사용하면 다음의 이익을 얻을 수 있습니다 .

- 원하는 문서 검색
- 지원 사례의 진도 제출 및 추적
- 지원 계약 관리
- HP 지원 연계 검색
- 가용한 서비스에 대한 정보를 검토
- 기타 고객과의 온라인 토론 참여
- 소프트웨어 교육에 대한 연구 및 등록

지원 영역의 대부분은 HP 허가증 사용자로 등록하여 로그인할 것을 요구합니다 . 지원 계약도 요구할 수 있습니다 . 액세스 레벨에 대한 자세한 정보를 찾으려면 다음의 URL 로 가십시오 .

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

HP 허가증 ID 를 등록하려면 다음의 페이지로 가십시오 .

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

1 장	개요	7
	버전 기록	7
	부가 정보 소스	7
2 장	패키지 설치	9
	원활한 설치를 위한 지침	9
	Interface Reporting ifEntry Datapipe 설치	10
	패키지 제거	11
3 장	데이터파이프 구성	13
	구성 태스크	13
	폴링 규칙 구성	14
	인터페이스 보고로 번들된 양식에 액세스하기	15
	등록 정보 데이터 내보내기 및 가져오기	16
	폴링 플래그 설정	16
	인터페이스 식별자 변경 (색인 다시 만들기 지원)	19
	다중 폴러에 대한 데이터파이프 구성	20
4 장	그룹 및 정책	23
	폴링 그룹	23
	규칙 기반 폴링 그룹	24
	폴링 정책	24
	IR_map_p	25
5 장	폴링된 객체 목록	27
	테이블 1: ifEntry 의 객체	27
	테이블 2: MIB-II 확장 객체	30
색인	33

개요

Interface Reporting ifEntry Datapipe 는 SNMP 관리 가능한 장치로부터 MIB-II ifEntry 및 ifxEntry 데이터를 수집합니다. 데이터를 수집하는 것 외에 Interface Reporting ifEntry Datapipe 는 다음의 기능을 제공합니다.

- 색인 다시 만들기 지원 (Interface Discovery Datapipe 로부터 파생)
- 인스턴스당 폴링이라고도 하는 규제된 인스턴스 폴링
- 값이 32 비트를 초과할 때마다 고용량 카운터
- NNM(Network Node Manager) 을 통한 폴링 구성의 선택적인 동기화

Interface Reporting ifEntry Datapipe 는 구성을 요구하지 않습니다. 3 장에서 설명되는 구성 옵션은 선택 사항입니다.

버전 기록

버전	RNS 버전 - 날짜	OVPI 버전 / 확장
1.1	RNS 4.0 - 2003 년 6 월	OVPI 4.5 및 4.6; Sybase 지원
2.0	RNS 5 - 2004 년 4 월	OVPI 5.0; Oracle 지원
2.0	RNS 6.0 - 2004 년 8 월	변경 없음
2.1	RNS 7.0 - 2004 년 11 월	인터페이스 동기화 확장 (OVPI 및 NNM) - Sybase 인터페이스 동기화 확장 (OVPI 및 NNM) - Oracle 폐기 , 오류 및 이용률 백분율을 계산하는 공식에 대한 부분 수정

부가 정보 소스

다음의 문서는 이 매뉴얼과 관련됩니다.

- *Interface Reporting ifEntry Datapipe Release Statement 2.1*

- *Interface Discovery Datapipe 2.1 User Guide*
- *Interface Reporting Report Pack 4.6 User Guide*
- *Frame Relay CPE Datapipe 4.0 User Guide*
- *RNS 7.0 Release Notes, 2004 년 11 월*

핵심 제품에서 실행하는 보고 솔루션에 대한 매뉴얼은 물론 핵심 제품, OVPI에 대한 매뉴얼은 다음의 사이트에서 다운로드할 수 있습니다.

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

지원 > 제품 매뉴얼을 선택하여 **제품 매뉴얼 검색** 페이지를 엽니다. OVPI에 대한 사용자 가이드는 **Performance Insight** 아래에 나열됩니다. 보고 솔루션, NNM SPI 및 NNM 상관기에 대한 사용자 가이드는 **보고 및 네트워크 솔루션** 아래에 나열됩니다.

보고 및 네트워크 솔루션 아래에 나열된 매뉴얼은 출판 연도 및 월을 지적합니다. 만약 매뉴얼이 개정되어 다시 발표되면 소프트웨어 버전 번호가 변경되지 않아도 출판 날짜는 변경될 것입니다. 정기적으로 업데이트된 사용자 가이드를 발표하므로 사용 가능한 최신 PDF가 아닌 이전 PDF를 사용하기 전에 이 사이트를 검색하여 업데이트해야 합니다.

패키지 설치

이 장은 다음 주제를 포함합니다.

- 원활한 설치를 위한 지침
- 패키지 관리자를 사용하여 패키지 설치
- 패키지 제거

원활한 설치를 위한 지침

RNS 7.0 CD에는 NNM 컴포넌트 및 OVPI 용 리포트 팩이 포함되어 있습니다. 설치용 OVPI 패키지를 선택할 때 CD의 설치 스크립트는 CD에서 모든 OVPI 패키지를 추출하고 시스템의 Packages 디렉토리에 결과를 복사합니다. 추출이 끝나면 설치 스크립트는 패키지 관리자 설치 방법을 시작하도록 프롬프트합니다.

만약 RNS 7.0 CD의 OVPI 패키지들을 이미 추출했다면 OVPI와 관련된 모든 패키지가 이미 시스템에 있으며 설치 가능합니다. 패키지 관리자를 시작하고 화면 지시에 따라 **Interface Reporting ifEntry Datapipe**를 설치합니다. 패키지 관리자를 사용하기 전에 다음의 지침을 검토합니다.

소프트웨어 전제 조건

Interface Reporting ifEntry Datapipe 2.1은 다음의 전제 조건을 갖습니다.

- OVPI 5.0
- OVPI 5.0에 사용 가능한 모든 서비스 팩
- Interface Discovery Datapipe 2.1

서비스 팩은 다음의 사이트에서 사용 가능합니다.

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

먼저 Interface Discovery Datapipe를 설치하거나 동시에 두 데이터파이프를 모두 설치할 수도 있습니다.

분산 환경

만약 시스템이 분산형이면 설치하는 더 복잡합니다. 폴링을 수행하는 모든 OVPI 서버에 **Interface Reporting ifEntry Datapipe** 를 설치하는지 확인합니다. 중앙 서버가 폴링을 수행하지 않는 경우에는 중앙 서버에 **Interface Reporting ifEntry Datapipe** 를 설치하지 마십시오.

버전 1.1 또는 2.0 으로부터 업그레이드

Interface Reporting ifEntry Datapipe 용 업그레이드 패키지는 없습니다. 만약 버전 1.1 또는 2.0 을 실행하고 있다면 첫 단계는 패키지 관리자를 시작하고 현재의 버전을 제거하는 것입니다. 그 다음 패키지 관리자를 다시 시작하고 버전 2.1 을 설치합니다.

Interface Reporting ifEntry Datapipe **설치**

RNS 7.0 CD 를 삽입하고 OVPI 솔루션 컴포넌트를 추출하기 위한 화면 지시를 따릅니다. Windows 에서 지시는 자동적으로 열리는 메인 메뉴에 나타나며 UNIX 에서는 루트 사용자로서 로그인하고 CD 를 마운트하며 (CD 가 자동적으로 마운트되지 않는 경우) CD 디렉토리의 최상위로 네비게이트하여 `./setup` 명령을 실행합니다.

OVPI 패키지가 이미 추출되어 시스템에 복사된 후에 설치 스크립트는 OVPI 를 시작하고 패키지 관리자를 시작합니다. 만약 CD 에서 패키지를 방금 추출했고 패키지 관리자 설치 마법사가 실행 중이면 **단계 4** 에서 이 절차를 시작합니다. 그렇지 않으면 **단계 1** 에서 시작합니다.

- 1 시스템에 로그인합니다. UNIX 시스템에서는 루트 사용자로서 로그인합니다.
- 2 OVPI 타이머를 정지하고 프로세스가 종료되기를 기다립니다.

Windows: **설정 > 제어판 > 관리 도구 > 서비스**를 선택합니다.

UNIX: 루트 사용자로서 다음 중 하나를 입력합니다.

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

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

- 3 Management Console에서 **도구 > 패키지 관리자**를 선택합니다. 패키지 관리자 환영 창이 열립니다.
- 4 **다음**을 클릭합니다. 패키지 위치 창이 열립니다.
- 5 **설치**를 클릭합니다. 기본 목적지 디렉토리를 확인하거나 필요하면 다른 디렉토리를 찾아봅니다.
- 6 **다음**을 클릭합니다. 보고서 배치 창이 열립니다. OVPI 사용자 이름과 암호를 입력합니다.
- 7 **다음**을 클릭합니다. 패키지 선택 창이 열립니다.
- 8 다음의 옆에 있는 체크 박스를 클릭합니다.

Interface_Reporting_ifEntry_Datapipe 2.1

IRifEntry_Sync_Interfaces

- 9 다음을 클릭합니다. 유형 발견 창이 열립니다. 기본값을 지웁니다.

 만약 **Interface Discovery Datapipe** 도 지금 설치하고 있다면 유형 발견 옵션에 대한 자세한 정보는 *Interface Discovery Datapipe User Guide* 를 참조하십시오.

- 10 다음을 클릭합니다. 선택 요약 창이 열립니다.

- 11 **설치**를 클릭합니다. 설치 진행 창이 열리며 설치를 시작합니다. 설치가 끝나면 패키지 설치 완료 메시지가 나타납니다.

- 12 **완료**를 클릭하여 관리 콘솔로 돌아갑니다.

 만약 **Interface Discovery Datapipe** 도 설치하고 있다면 **OVPI** 타이머를 다시 시작하기 전에 적합한 인터페이스 식별자 색인별 그룹에 장치를 다시 할당하려고 할 수도 있습니다. 이 단계는 *Interface Discovery Datapipe User Guide* 에서 설명됩니다.

- 13 필요하면 장치를 적합한 인터페이스 식별자 그룹에 다시 할당합니다.

- 14 **OVPI** 타이머를 다시 시작합니다.

Windows: **설정 > 제어판 > 관리 도구 > 서비스**를 선택합니다.

UNIX: 루트 사용자로서 다음 중 하나를 입력합니다.

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

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

Interface Reporting ifEntry Datapipe 는 현재 시스템에 설치됩니다.

패키지 제거

Interface Reporting ifEntry Datapipe 를 설치 해제하려면 아래의 단계를 따릅니다.

- 1 시스템에 로그인합니다. **UNIX** 시스템에서는 루트 사용자로서 로그인합니다.
- 2 **OVPI** 타이머를 정지하고 프로세스가 종료되기를 기다립니다.
- 3 패키지 관리자를 시작합니다. 패키지 관리자 설치 마법사가 열립니다.
- 4 패키지 제거에 대한 화면 지시에 따르며 제거할 **Interface Reporting ifEntry Datapipe** 를 선택합니다. 설치 해제가 끝나면 패키지 제거 완료 메시지가 나타납니다.
- 5 **완료**를 클릭하여 관리 콘솔로 돌아갑니다.
- 6 **OVPI** 타이머를 다시 시작합니다.

데이터파이프 구성

이 장은 다음의 주제들을 포함합니다.

- 구성 태스크 및 그것을 완성하는 방법
- 폴링 규칙 구성
- 인터페이스 보고 4.5 로 변들된 양식에 액세스하기
- 등록 정보 데이터 내보내기 및 가져오기
- 폴링 플래그 설정
- 인터페이스 식별자 변경
- 다중 폴러에 대한 데이터파이프 구성

구성 태스크

다음의 테이블은 각 구성 태스크를 태스크를 완성하기 위한 하나 이상의 방법에 맵핑합니다.

태스크	완성하는 방법
폴링 규칙 구성	구성 도구
데이터 필터링 모드 설정	인터페이스 보고 관리 (양식)
준비 모드 설정	인터페이스 보고 관리 (양식)
메시지 레벨 설정	인터페이스 보고 관리 (양식)
등록 정보 데이터 내보내기 및 가져오기	IR 구성 도구
폴링 플래그 설정	<ol style="list-style-type: none"> 1 인터페이스 폴링 플래그 관리 (양식) 2 폴링 정책 관리자 3 IR 리포트 팩에 대한 등록 정보 가져오기 4 폴링을 NNM 와 동기화 5 사용자 정의 SQL 스크립트

태스크	완성하는 방법
프로토콜 모니터링 사용 안 함	프로토콜 기본값 변경 (양식)
인터페이스 식별자 변경	<ol style="list-style-type: none"> 1 Interface Discovery Datapipe 2 폴링 정책 관리자
다중 폴러에 대한 데이터파이프 구성	<ol style="list-style-type: none"> 1 OVPI group_manager 2 OVPI collection_manager 3 폴링 정책 관리자

폴링 규칙 구성

폴링은 다음의 5 개 규칙 중 하나에 맞추어 수행됩니다.

(1) 모든 활성 인터페이스 폴링

ifAdminStatus 는 활성으로 설정된 모든 인터페이스를 폴링합니다.

(2) 모든 인터페이스 폴링

ifAdminStatus 또는 프로토콜에 관계없이 모든 인터페이스를 폴링합니다. (이것은 인터페이스 보 고 ifEntry Datapipe 가 설치되었을 때의 기본 규칙입니다.)

(3) 모니터링되는 프로토콜로 모든 활성 인터페이스 폴링

기본값입니다. ifAdminStatus 가 활성으로 설정되고 KIR_protocol_config 테이블에서 프로토콜 (ifType) 이 모니터링에 대해 구성되는 모든 인터페이스를 폴링합니다. 모니터링을 요구하지 않는 프로토콜에 대한 모니터링을 사용하지 않을 수 있습니다.

▶ 기본값은 모니터링되는 프로토콜로 모든 활성 인터페이스 폴링입니다.

(4) 모니터링되는 프로토콜로 모든 인터페이스 폴링

KIR_protocol_config 테이블에서 프로토콜 (ifType) 이 모니터링에 대해 구성되는 모든 인터페이스 를 폴링합니다. 모니터링을 요구하지 않는 프로토콜에 대한 모니터링을 금지할 수 있습니다.

(5) 플래그된 인터페이스만 폴링

폴링 플래그 설정을 갖는 인터페이스만을 폴링합니다. 이 설정은 개별적인 인터페이스가 수집에 대 해 준비되는 것을 활성화합니다. 이 설정을 사용하여 NNM 의 " 관리되지 않은 인터페이스 " 기능과 의 통합을 위한 폴링을 제어합니다.

폴링 규칙을 구성하려면 아래의 단계를 따릅니다.

- 1 시스템에 로그인합니다. UNIX 에서는 루트 사용자로서 로그인합니다.
- 2 다음의 명령줄로부터 IR 구성 도구를 시작합니다.

UNIX:

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

Windows:

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

IR 구성 도구 메인 메뉴가 열립니다.

- 3 옵션 1, 폴링 구성을 선택하고 **입력**을 누릅니다.
폴링 구성 메뉴가 열립니다.
- 4 옵션 1, 폴링 규칙 설정을 선택하고 **입력**을 누릅니다.
폴링 규칙 설정 메뉴가 열리며 현재의 폴링 규칙과 규칙이 마지막으로 변경된 날짜 및 시간을 표시합니다.
- 5 요구되는 폴링 규칙에 대한 옵션을 선택하고 **입력**을 누릅니다.
폴링 규칙을 구현하는 데이터베이스 보기를 다시 빌드할 때 메시지를 표시한 후 폴링 규칙 설정 메뉴가 다시 열리며 새 규칙을 보여줍니다.
- 6 **E**를 선택하고 **입력**을 눌러 이전 메뉴에 리턴합니다.
폴링 구성 메뉴가 열립니다.
- 7 **E**를 선택하고 **입력**을 눌러 이전 메뉴에 리턴합니다.
IRCT 메인 메뉴가 열립니다.
- 8 옵션을 선택하거나 **E**를 선택하고 **입력**을 눌러 종료합니다.



만약 "플래그된 인터페이스만 폴링" 규칙을 적용한다면 폴링하려는 인터페이스에 대한 폴링 플래그도 설정해야 합니다.

인터페이스 보고로 번들된 양식에 액세스하기

인터페이스 보고 4.5 로 번들된 양식에 액세스하려면 아래의 단계를 따릅니다.

- 1 Management Console 을 시작합니다.
- 2 **객체** 아이콘을 클릭합니다. 객체 / 등록 정보 관리 창이 열립니다.
- 3 장치를 선택합니다.

다음의 양식은 **객체별 태스크** 아래에 나열됩니다.

- 프로토콜 기본값 변경



기본적으로 모든 프로토콜 (ifTypes) 이 모니터링됩니다. 이것은 데이터가 모든 프로토콜에 대해 수집된다는 것을 의미합니다. 만약 특정 프로토콜이 필요 없다면 프로토콜 기본값 변경 양식을 사용하여 수집 및 보고를 사용하지 않도록 설정합니다.

- 인터페이스 변경
- 인터페이스 고객 변경

- 인터페이스 설명 변경
- 인터페이스 폴링 플래그 관리

다음의 양식은 **일반 태스크** 아래에 나열됩니다.

- 인터페이스 보고 관리

이 양식에 대한 자세한 정보는 *인터페이스 보고 Report Pack 4.5 User Guide* 에서 6 장 변경 양식 사용을 참조하십시오.

등록 정보 데이터 내보내기 및 가져오기

IR 구성 도구는 인터페이스 보고 리포트 팩으로부터 기존 등록 정보 데이터를 내보내고 인터페이스 보고 리포트 팩에 새 등록 정보 데이터를 가져오는 것을 쉽게 합니다. IR 구성 도구를 사용하여 등록 정보 데이터를 내보내거나 가져올 때 데이터 파일은 인터페이스 보고 리포트 팩이 갖고 있는 등록 정보 데이터의 기본 디렉토리에 내보내지고 가져오게 되는데 기본 디렉토리는 다음과 같습니다.

`$DPIPE_HOME/data/PropertyData`

등록 정보 데이터를 내보내거나 가져오려면 아래의 단계를 따릅니다.

- 1 시스템에 로그인합니다. UNIX에서는 루트 사용자로서 로그인합니다.
- 2 다음의 명령줄로부터 IR 구성 도구를 시작합니다.

UNIX:

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

Windows:

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

IRCT 메인 메뉴가 열립니다.

- 3 옵션 2, 인터페이스 보고 등록 정보 데이터 내보내기 또는 가져오기를 선택하고 **입력**을 누릅니다.

등록 정보 내보내기 및 가져오기 메뉴가 열립니다.

- 4 다음 중 하나를 수행합니다.
 1. **인터페이스 보고 정보 데이터 내보내기**를 선택하여 내보냅니다.
 2. **인터페이스 보고 등록 정보 데이터 가져오기**를 선택하여 가져옵니다.

- 5 끝나면 **E**를 선택하고 **입력**을 눌러 이전 메뉴로 리턴합니다.

IR 구성 도구 메인 메뉴가 열립니다.

- 6 옵션을 선택하거나 **E**를 선택하고 **입력**을 눌러 종료합니다.

폴링 플래그 설정

폴링 플래그를 설정하려면 다음의 방법 중 하나를 사용합니다.

- 인터페이스 폴링 플래그 관리 (양식)
- 폴링 정책 관리자
- 인터페이스 보고 리포트 팩의 등록 정보 가져오기
- NNM 에서 “관리되지 않은 인터페이스” 폴링 구성과의 동기화

 폴링 플래그를 설정하는 이 방법은 Sybase 에서만 OVPI 에 적용합니다.

- 사용자 정의 SQL 스크립트 생성

폴링 정책 관리자를 사용하여 폴링 플래그 설정

플래그된 인터페이스만 폴링 규칙이 활성화되었을 때 인터페이스 보고 ifEntry Datapipe 는 메인 인터페이스 등록 정보 테이블 K_IFEntry_Disc.poll_flag_ifEntry 에서 설정한 폴링 플래그를 갖는 인터페이스로부터만 통계를 수집합니다. 폴링 정책 관리자를 사용하여 폴링 플래그를 수정하려면 아래의 단계를 따릅니다.

- 1 어느 장치에서 어느 인터페이스가 폴링되고 어느 장치에서 어느 인터페이스가 폴링되지 않을지를 확정합니다.
- 2 시스템에 로그인합니다. UNIX 에서는 루트 사용자로서 로그인합니다.
- 3 폴링 정책 관리자를 시작합니다.
- 4 **편집 > 설명 ...** 을 선택합니다. 설명 편집 창이 열립니다.
- 5 등록 정보 그룹 K_IFEntry_Disc를 선택하고 **규칙:편집...** 을 클릭합니다. 규칙 생성 창이 열립니다.
- 6 규칙을 선택하거나 입력합니다. 예를 들어 고속 인터페이스만을 리턴하려면 interface_speed >= 100000000 을 사용하고 **확인** 을 클릭합니다.
- 7 **결과 인스턴스 보기** 를 클릭합니다. 하위 창에 인스턴스가 나타날 것입니다.
- 8 리턴된 목록으로부터 폴링될 모든 인터페이스를 선택하고 **인스턴스 값 편집 ...** 을 클릭합니다. 인스턴스 편집 창이 열립니다.
- 9 poll_flag_ifEntry 열을 선택하고 값 1 을 입력하고 **적용** 을 클릭한 다음 **닫기** 를 클릭하여 설명 편집 창으로 리턴합니다.
- 10 **결과 인스턴스 보기** 를 클릭합니다. 하위 창에 업데이트된 인스턴스가 나타날 것입니다.
- 11 **닫기** 를 클릭합니다.
- 12 만약 플래그된 인터페이스만 폴링 규칙이 이미 활성화되지 않았다면 지금 그것을 활성화합니다.

등록 정보 가져오기를 사용하여 폴링 플래그 설정

폴링된 인터페이스만 폴링 규칙이 활성화되었을 때 인터페이스 보고 **ifEntry Datapipe** 는 메인 인터페이스 등록 정보 테이블에서 설정된 폴링 플래그를 갖는 인터페이스로부터만 통계를 수집합니다. 인터페이스 보고 등록 정보 가져오기를 사용하여 폴링 플래그를 설정하려면 아래의 단계를 따릅니다.

▶ 만약 데이터 필터링 모드가 활성화되어 준비된 인터페이스에 대한 데이터만 수집되는 경우 폴링 플래그는 등록 정보 가져오기 파일에 포함된 각 인터페이스에 자동적으로 설정됩니다.

- 1 어느 장치에서 어느 인터페이스가 폴링되고 어느 장치에서 어느 인터페이스가 폴링되지 않았는지를 확인합니다.
- 2 시스템에 로그인합니다. UNIX에서는 루트 사용자로서 로그인합니다.
- 3 폴링될 인터페이스를 갖는 새 인터페이스 보고 등록 정보 가져오기 파일을 생성하거나 만약 인터페이스가 이미 발견되었다면 등록 정보 파일을 내보내고 그것을 편집하여 원하는 인터페이스만을 포함합니다.
- 4 인터페이스 보고 관리 양식을 사용하여 데이터 필터링 모드가 온인지 오프인지 검사합니다. 만약 데이터 필터링 모드가 이미 활성화되지 않았다면 지금 그것을 활성화합니다.
- 5 단계 3에서 생성한 인터페이스 보고 등록 정보 데이터를 가져옵니다.
- 6 만약 폴링된 인터페이스만 폴링 규칙이 이미 활성화되지 않았다면 지금 그것을 활성화합니다.

NNM 와 동기화하여 폴링 플래그 설정

▶ 폴링 플래그를 설정하는 이 방법은 Sybase 에서만 OVPI 에 적용합니다.

폴링된 인터페이스만 폴링 규칙이 활성화되었을 때 인터페이스 보고 **ifEntry Datapipe** 는 메인 인터페이스 등록 정보 테이블에서 설정된 폴링 플래그를 갖는 인터페이스로부터만 통계를 수집할 것입니다. 이 플래그는 NNM 에서 “관리되지 않는 인터페이스” 폴링 구성과 동기화하여 설정될 수 있습니다. NNM 는 두 가지 방법 즉 사용자 인터페이스를 사용하거나 이러한 목적으로 설계된 특별 추가, 규칙 기반 도구를 사용하여 관리되지 않는 인터페이스를 정의하는 것을 활성화합니다.

IRifEntry Sync Interfaces 패키지를 설계하여 NNM 로부터 관리되는 및 관리되지 않는 인터페이스 목록을 업로드하고 관리되는 인터페이스에 대해서는 폴링 플래그를 설정하고 관리되지 않는 인터페이스에 대해서는 플래그를 지웁니다. 이 패키지는 통합 모듈의 **Device Sync** 패키지에 의거합니다. 두 패키지를 모두 설치하는 방법에 대한 자세한 내용은 **NNM / Performance Insight 통합 모듈 2.0 User Guide** 를 참조하십시오.

IRifEntry Sync Interfaces 패키지를 설치할 때 아래의 이벤트가 일어납니다.

- 관리되는 및 관리되지 않는 인터페이스의 완전 목록이 NNM 로부터 다운로드됩니다.
- 인터페이스 폴링 플래그가 (NNM 구성에 기반하여) 설정됩니다.
- **OVPI Timer** 엔트리를 구성하여 다음을 수행합니다.

- 이전 24 시간 동안 관리 상태의 변경을 수행하는 모든 인터페이스의 매 일 동기화
- 한 달에 한 번 완전 다시 동기화

NNM 와 동기화하여 폴링 플래그를 설정하려면 아래의 단계를 따릅니다.

- 1 만약 통합 모듈로 번들된 패키지가 NNM 및 OVPI 에 이미 설치되지 않았다면 그것들을 설치합니다.
- 2 만약 IRifEntry Sync Interfaces 패키지가 이미 설치되지 않았다면 지금 그것을 설치합니다. 2 장, 패키지 설치에서 제공된 절차를 따릅니다.
- 3 만약 플래그된 인터페이스만 폴링 규칙이 이미 활성화되지 않았다면 지금 그것을 활성화합니다.

사용자 정의 SQL 스크립트 생성

만약 폴링 구성이 준비 또는 구성 관리 시스템에서 보유되면 폴링 플래그 (K_IFEntry_Disc.poll_flag_ifEntry) 를 설정하도록 사용자 정의 SQL 스크립트를 개발합니다.

인터페이스 식별자 변경 (색인 다시 만들기 지원)

Interface Discovery Datapipe 는 다음의 대체 식별자를 제공합니다.

- ifDescr- index-by-ifDescr 폴링 그룹에 의해 지원됨
- ifName- index-by-ifName 폴링 그룹에 의해 지원됨
- ifAlias- index-by-ifAlias 폴링 그룹에 의해 지원됨

색인 다시 만들기를 활성화하려면 장치에 대해 유일하고 지속적인 대체 식별자 (하나의 대체 식별자) 를 선택합니다.

 하나의 장치를 둘 이상의 그룹에 할당하지 마십시오 . 장치당 하나의 대체 식별자만이 활성화됩니다.

Interface Discovery Datapipe 는 특정 장치 유형을 대체 인터페이스 식별자에 자동적으로 할당하는 벤더별 OVPI 유형 탐색 파일을 포함하므로 일부 선택 작업이 이미 완료됩니다. 다음의 테이블은 장치 유형과 미리 할당된 식별자 목록을 포함합니다.

장치 유형	OVPI 유형 탐색에 의해 할당된 식별자
Cisco Catalyst 스위치	ifName
Cisco 라우터	ifDescr
HP ProCurve 스위치 (800T 는 제외)	ifName

index-by-ifIndex 그룹은 기본 그룹입니다. 이 그룹은 index-by-ifAlias, index-by-ifDescr 또는 index-by-ifName 그룹에 없는 GENMIB2IF_Datapipe 그룹의 모든 장치를 포함합니다.

인터페이스 식별자를 변경하려면 아래의 단계를 따릅니다.

- 1 각 장치 유형에 대해 인터페이스 식별자를 선택합니다.
- 2 적합한 색인별 그룹에 해당 유형의 각 장치를 할당합니다.
 - a 폴링 정책 관리자를 시작합니다.
 - b **편집 > 폴링 그룹 ...** 을 선택합니다.
 - c 최상위 창에서는 **같은 유형의 모든 노드**를 선택하고 최하위 창에서는 요구된 색인별 그룹을 선택합니다.
 - d **편집**을 클릭합니다.
 - e 왼쪽 창에서 이 그룹에 할당될 노드를 선택하고 >> 를 클릭하여 그것들을 그룹으로 이동합니다. 완료될 때 **확인**을 클릭합니다.
- 3 필요하면 기타 장치 유형 및 색인별 그룹에 대해 위의 절차를 반복합니다.

만약 장치가 색인별 그룹에 할당되었지만 식별자가 유일하지 않으면 다음의 메시지와 유사한 중복 식별자 메시지가 경향 로그에 삽입될 것입니다.

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

다중 폴러에 대한 데이터파이프 구성

인터페이스 보고 ifEntry Datapipe 는 다중 폴러로 사용하기 위한 추가적인 구성을 요구합니다. 표준 설치하는 단일 폴러에 대한 4 개의 그룹 및 4 개의 폴링 정책을 구성합니다. 다중 폴러로 사용하기 위한 그룹 및 폴링 정책의 추가적인 설정이 추가적인 각 폴러에 대해 구성되어야 합니다.

데이터파이프 패키지는 두 개의 폴러에 대한 추가적인 그룹 및 수집 정책 파일을 포함합니다. 만약 하나의 데이터베이스에 세 개 이상의 폴러에 대한 데이터파이프를 구성해야 한다면 그룹 파일 및 정책의 추가적인 설정을 생성하기 위해 아래의 단계를 따릅니다.

- 1 OVPI 관리자 권한으로 로그인합니다 (trendadm).
- 2 인터페이스 보고 ifEntry Datapipe 패키지 디렉토리에 네비게이트합니다.
- 3 다음의 명령을 실행합니다.

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

- 4 다음의 명령을 실행합니다.

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

- 5 폴링 정책 관리자를 시작하고 **편집 > 데이터파이프 설치 ...** 를 선택합니다.
- 6 **생성 ...** 을 클릭하여 새 폴러를 생성합니다.
- 7 각 폴러의 호스트 이름과 (선택적으로) IP 주소를 추가하고 **확인**을 클릭합니다.
- 8 모든 폴러가 추가되었을 때 **닫기**를 클릭하여 데이터파이프 설치 편집 창을 종료합니다.
- 9 왼쪽 창에서 **모두**를 클릭하여 모든 폴링 정책을 표시합니다.
- 10 최초의 if*_Collection 정책 (이름 끝에 _poller1 또는 _poller2 가 없는 것들) 을 삭제합니다.

- 11 정책을 더블 클릭하여 새 if*_Collection_poller1 정책을 수정합니다.
폴러 1의 호스트 이름이 되도록 **할당된 폴링** 설정을 변경합니다.
- 12 정책을 더블 클릭하여 새 if*_Collection_poller2 정책을 수정합니다.
폴러 2의 호스트 이름이 되도록 **할당된 폴링** 설정을 변경합니다.
- 이 시점에서 15분 정책은 다음의 테이블의 정책과 유사하게 보일 것입니다.

정책 이름	수집될 데이터의 테이블 별명 이름	수집할 그룹 이름	폴링
ifAlias_Collection_poller1	xV_IRifEntry_ifAlias	ifAlias-poller1	폴러 1의 호스트 이름
ifDescr_Collection_poller1	xV_IRifEntry_ifDescr	ifDescr-poller1	폴러 1의 호스트 이름
ifIndex_Collection_poller1	xV_IRifEntry_ifIndex	ifIndex-poller1	폴러 1의 호스트 이름
ifName_Collection_poller1	xV_IRifEntry_ifName	ifName-poller1	폴러 1의 호스트 이름
ifAlias_Collection_poller2	xV_IRifEntry_ifAlias	ifAlias-poller2	폴러 2의 호스트 이름
ifDescr_Collection_poller2	xV_IRifEntry_ifDescr	ifDescr-poller2	폴러 2의 호스트 이름
ifIndex_Collection_poller2	xV_IRifEntry_ifIndex	ifIndex-poller2	폴러 2의 호스트 이름
ifName_Collection_poller2	xV_IRifEntry_ifName	ifName-poller2	폴러 2의 호스트 이름

- 13 상위 창에서 **편집 > 폴링 그룹... > 같은 보기의 모든 노드**를 선택하고 하위 창에서 폴러1을 선택합니다. 폴러 1에 의해 수집될 노드들을 폴러 1 보기에 할당합니다.
- 14 상위 창에서 **편집 > 폴링 그룹... > 같은 보기의 모든 노드**를 선택하고 하위 창에서 폴러2를 선택합니다. 폴러 2에 의해 수집될 노드들을 폴러 2 보기에 할당합니다.

그룹 및 정책

이 장은 다음 주제를 포함합니다.

- 폴링 그룹
- 규칙 기반 폴링 그룹
- 폴링 정책
- IR_map_p 저장 프로시저

폴링 그룹

Interface Reporting `ifEntry Datapipe` 는 Interface Discovery `Datapipe` 가 사용하는 것과 같은 OVPI 폴링 "유형" 그룹을 사용합니다. 폴링 그룹은 색인 다시 만들기 지원을 가능하게 합니다. 폴링 그룹은 Interface Discovery `Datapipe` 가 설치될 때 생성됩니다.

폴링 그룹	설명
GENMIB2IF_Datapipe	MIB-II 를 지원하는 모든 노드의 최상위 레벨 그룹입니다.
index-by-ifAlias	식별자로서 <code>ifAlias</code> 를 사용하는 모든 노드를 포함하는 GENMIB2IF_Datapipe 그룹의 서브세트입니다.
index-by-ifDescr	식별자로서 <code>ifDescr</code> 를 사용하는 모든 노드를 포함하는 GENMIB2IF_Datapipe 그룹의 서브세트입니다.
index-by-ifIndex	<code>index-by-ifAlias</code> , <code>index-by-ifDescr</code> 또는 <code>index-by-ifName</code> 그룹에 속하지 않는 모든 장치를 포함하는 GENMIB2IF_Datapipe 그룹의 서브세트입니다. 장치가 다른 그룹에 할당되지 않는 경우 그것은 인터페이스 식별자로서 <code>ifIndex</code> 를 기본값으로 할 것입니다. 이 그룹을 편집할 필요는 없습니다.
index-by-ifName	식별자로서 <code>ifName</code> 을 사용하는 모든 노드를 포함하는 GENMIB2IF_Datapipe 그룹의 서브세트입니다.

index-by-ifIndex 그룹이 기본값입니다. 이 그룹은 폴링을 제어하는 것이 아니라 어느 장치가 인터페이스 식별자로서 ifIndex 를 사용하는가를 보여줍니다. 이 그룹은 한 시간에 한번 씩 시스템에 의해 자동적으로 다시 채워지기 때문에 다른 색인별 그룹에 다시 할당된 어떤 장치가 index-by-ifIndex 에서 즉시 없어지지 않을 수도 있습니다. 그러나 index-by-ifIndex 에 대한 데이터 수집이 GENMIB2IF_Datapipe 그룹의 장치에 기반되기 때문에 장치는 그의 새 색인별 그룹에서만 폴링 될 것입니다.

규칙 기반 폴링 그룹

5 개 기본 폴링 그룹 외에 Interface Reporting ifEntry Datapipe 는 기본 그룹으로부터 파생될 뿐 아니라 규칙 기반 폴링 규칙도 포함하는 4 개 그룹을 추가합니다.

그룹 이름	데이터베이스 보기	설명
ifAlias-all	KV_IFEntry_by_ifAlias	적용된 폴링 규칙을 갖는 index-by-ifAlias 그룹에서 파생되었습니다.
ifDescr-all	KV_IFEntry_by_ifDescr	적용된 폴링 규칙을 갖는 index-by-ifDescr 그룹에서 파생되었습니다.
ifIndex-all	KV_IFEntry_by_ifIndex	적용된 폴링 규칙을 갖는 index-by-ifAlias, index-by-ifDescr 또는 index-by-ifName 에 없는 GENMIB2IF_Datapipe 그룹의 장치에서 파생되었습니다.
ifName-all	KV_IFEntry_by_ifName	적용된 폴링 규칙을 갖는 index-by-ifName 그룹에서 파생되었습니다.

각 그룹에서 폴링될 인터페이스 목록을 보려면 SQL 을 사용하여 위 테이블의 데이터베이스 보기에서 선택합니다.

폴링 정책

Interface Reporting ifEntry Datapipe 의 표준 설치는 각 색인별 그룹에 하나씩 즉 4 개의 15 분 폴링 정책의 집합을 허용합니다. 다중 폴러에 대한 지원을 허용하지 않는 경우 이 정책을 변경할 필요가 없습니다.

정책 이름	수집될 데이터의 테이블 별명 이름	수집할 그룹 이름
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 는 Interface Discovery Datapipe 등록 정보 테이블 K_IFEntry_Disc 에 poll_flag_ifEntry 라는 플래그 열을 추가합니다. 이 플래그의 준비는 규제된 인스턴스 폴링을 제어할 것입니다.

IR_map_p

Interface Reporting ifEntry Datapipe 는 Interface Reporting 리포트 팩에 수집된 데이터를 전송하기 위해 IR_map_p 라는 저장 프로시저를 사용합니다. IR_map_p 는 원시 비율 전환 끝에서 호출됩니다. IR_map_p 및 전환 절차는 위 4 개의 각 폴링 정책의 끝에서 실행합니다. 그것들이 실행 중인 동안 이러한 절차는 데이터가 유실되지 않도록 테이블을 잠급니다. 결과적으로 원시 비율 저장 프로시저의 하나 이상의 인스턴스들이 인스턴스가 이미 실행 중임을 나타내는 잠금 오류를 발생할 것입니다. 다음의 내용은 오류 메시지의 예제입니다.

```
"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 는 Interface Reporting 리포트 팩에 전송된 행 수를 갖는 표준 정보 메시지도 생성합니다. 예를 들어:

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


폴링된 객체 목록

다음의 테이블은 폴링된 객체의 이름, 식별자 및 설명을 제공합니다. **ifEntry** MIB 테이블에서 테이블 1 (아래)은 객체 ID에 의해 구성된 객체의 목록을 제공합니다. 테이블 2 (30 페이지의)는 **ifXTable**로도 알려진 MIB-II 확장 테이블에서 객체 ID에 의해 구성된 객체의 목록을 제공합니다.

테이블 1: ifEntry 의 객체

MIB 객체 이름	OID	설명
ifIndex	1.3.6.1.2.1.2.2.1.1	1 과 ifNumber 값 사이 범위에 있는 각 인터페이스에 대한 유일한 값입니다. 각 인터페이스의 값은 적어도 엔터티의 네트워크 관리 시스템의 한 재초기화로부터 다음 재초기화까지 변하지 말아야 합니다.
ifDescr	1.3.6.1.2.1.2.2.1.2	인터페이스에 대한 정보를 포함하는 텍스트 문자열입니다. 이 문자열은 제조업체의 이름, 제품 이름 및 하드웨어 인터페이스 버전을 포함해야 합니다.
ifType	1.3.6.1.2.1.2.2.1.3	프로토콜 스택의 네트워크 레이어 바로 아래의 물리적 / 링크 프로토콜 (들)에 따라 구별되는 인터페이스 유형입니다.
ifMtu	1.3.6.1.2.1.2.2.1.4	인터페이스에 대해 송신 / 수신할 수 있는 옥텟으로 지정된 제일 큰 데이터그램의 크기입니다. 네트워크 데이터그램을 전송하는 데 사용되는 인터페이스에 대해서는 이것이 인터페이스에 송신할 수 있는 제일 큰 데이터그램의 크기입니다.
ifSpeed	1.3.6.1.2.1.2.2.1.5	초당 비트 수의 인터페이스 현재 대역 폭의 추정입니다. 대역 폭이 변하지 않은 인터페이스에 대해 또는 정확한 추정을 할 수 없는 인터페이스에 대해 이 객체는 공칭 대역 폭을 포함해야 합니다.

MIB 객체 이름	OID	설명
ifPhysAddress	1.3.6.1.2.1.2.2.1.6	프로토콜 스택의 네트워크 레이어 바로 아래 프로토콜 레이어에서의 인터페이스 주소입니다. 그러한 주소를 가지지 않는 인터페이스 (예를 들어 직렬 회선)에 대해 이 객체는 0 길이의 옥텟 문자열을 포함해야 합니다.
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	원하는 인터페이스의 상태입니다. 테스트 상태에서 운용 패킷들은 통과될 수 없습니다. 상태는 업 (1), 다운 (2), 테스트 (3) 입니다.
ifOperStatus	1.3.6.1.2.1.2.2.1.8	인터페이스의 현재 운용 상태입니다. 테스트 상태에서 운용 패킷들은 통과될 수 없습니다. 상태는 업 (1), 다운 (2), 테스트 (3) 입니다.
ifLastChange	1.3.6.1.2.1.2.2.1.9	인터페이스가 그것의 현재 운용 상태를 입력했을 때의 sysUpTime 값입니다. 만약 현재 상태로 로컬 네트워크 관리 서버 시스템의 마지막 재초기화 전에 입력되었다면 이 객체는 영 값을 포함합니다.
IfInOctets	1.3.6.1.2.1.2.2.1.10	프레임 문자들을 포함하여 인터페이스에 수신된 옥텟의 총 개수입니다.
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	상위 레이어 프로토콜에 전달된 서브 네트워크 - 유니캐스트 패킷의 개수입니다.
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	상위 레이어 프로토콜에 전달된 유니캐스트가 아닌 (예를 들어 서브 네트워크 - 브로드캐스트 또는 서브 네트워크 - 멀티캐스트) 패킷의 개수입니다.
ifInDiscards	1.3.6.1.2.1.2.2.1.13	상위 레이어 프로토콜에 전달되는 것을 방지하기 위해 검출한 오류가 하나도 없더라도 폐기되도록 선택된 인바운드 패킷의 개수입니다. 그러한 패킷을 폐기하는 한 가지 가능한 이유는 버퍼 공간을 해제하는 것일 수 있습니다.
ifInErrors	1.3.6.1.2.1.2.2.1.14	상위 레이어 프로토콜에 전달되는 것을 방지하는 오류들을 포함한 인바운드 패킷의 개수입니다.
ifUnknownProtos	1.3.6.1.2.1.2.2.1.15	알려지지 않거나 지원되지 않는 프로토콜때문에 폐기된 인터페이스를 통해 수신된 패킷의 개수입니다.
ifOutOctets	1.3.6.1.2.1.2.2.1.16	프레이밍 문자들을 포함하여 인터페이스에서 송신되는 옥텟의 총 개수입니다.
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	폐기되었거나 송신되지 않은 것들을 포함하여 상위 레벨 프로토콜이 요청한 서브 네트워크 - 유니캐스트 주소로 전송되는 패킷의 총 개수입니다.

MIB 객체 이름	OID	설명
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	폐기되었거나 송신되지 않은 것들을 포함하여 요청된 상위 레벨 프로토콜이 유니캐스트가 아닌 (예를 들어 서브 네트워크 - 브로드캐스트 또는 서브 네트워크 - 멀티캐스트) 주소로 전송한 패킷의 총 개수입니다.
ifOutDiscards	1.3.6.1.2.1.2.2.1.19	전송되는 것을 방지하기 위해 검출한 오류가 하나도 없더라도 폐기되도록 선택된 아웃바운드 패킷의 개수입니다. 그러한 패킷을 폐기하는 하나의 가능한 이유는 버퍼 공간을 해제하는 것일 수 있습니다.
ifOutErrors	1.3.6.1.2.1.2.2.1.20	오류때문에 전송될 수 없었던 아웃바운드 패킷의 개수입니다.
ifOutQLen	1.3.6.1.2.1.2.2.1.21	출력 패킷 큐 (패킷에서)의 길이입니다.
ifSpecific	1.3.6.1.2.1.2.2.1.22	인터페이스를 인식하는 데 사용되는 특정 매체별 MIB 정의에 대한 참조입니다. 예를 들어 만약 인터페이스가 이더넷에 의해 인식되었다면 이 객체의 값은 이더넷 특정 객체를 정의하는 문서를 참조합니다. 만약 이 정보가 나타나지 않으면 값은 구문적으로 유효한 객체 식별자인 OBJECT IDENTIFIER { 0 0 } 으로 설정되며 ASN.1 및 BER의 구현에 따르는 것들은 이 값을 생성하여 인지할 수 있어야 합니다.

테이블 2:MIB-II 확장 객체

MIB 객체 이름	OID	설명
ifName	1.3.6.1.2.1.31.1.1.1.1	인터페이스의 텍스트 이름입니다. 이 객체의 값은 로컬 장치에 의해 할당된 것과 같은 인터페이스의 이름이어야 하며 장치의 콘솔에 입력된 명령으로 사용하는 데 적합해야 합니다. 이것은 장치의 인터페이스 이름 달기 구문에 따라 “1e0” 또는 간단한 포트 번호와 같은 텍스트 이름일 수도 있습니다. 만약 ifTable 의 몇개의 엔트리가 장치에 의해 명명된 것과 같은 단일 인터페이스를 함께 나타낸다면 각 엔트리는 같은 값의 ifName 을 가질 것입니다. 만약 로컬 이름이 없거나 이 객체를 적용할 수 없다면 이 객체는 0 길이 문자열을 포함합니다.
ifAlias	1.3.6.1.2.1.31.1.1.1.18	이 객체는 네트워크 관리자에 의해 지정된 것과 같은 인터페이스의 별명 이름이며 인터페이스에 대해 비휘발성 “핸들” 을 제공합니다. 인터페이스의 첫 번째 인스턴스에서 그 인터페이스와 연관된 ifAlias 의 값은 0 길이 문자열입니다. 네트워크 관리 설정 조작을 통해 ifAlias 의 인스턴스에 값이 기록될 때 네트워크 관리 시스템이 재초기화 / 리부팅한다 해도 그 인터페이스가 인스턴스화되어 있는 한 에이전트가 같은 인터페이스와 연관된 ifAlias 인스턴스의 제공된 값을 유지해야 합니다. 네트워크 관리자가 WAN 인터페이스에 대해 이 객체에 저장할 값의 예제는 (Telco 의) 인터페이스 서킷 번호 / 식별자입니다. 일부 에이전트는 ifType 의 특정 값을 가지는 인터페이스에 대해서만 쓰기 - 액세스를 지원할 수도 있습니다. 이 객체에 대한 쓰기 액세스를 지원하는 에이전트는 비휘발성 저장소에 값을 보관할 것을 요구하지만 다른 인터페이스에 대한 현재 값에 의해 이미 점유된 저장소의 크기에 따라 새 값의 길이를 제한할 수도 있습니다.
ifHCInOctets	1.3.6.1.2.1.31.1.1.1.6	프레임 문자들을 포함하여 인터페이스에서 수신된 옥텟의 총 개수입니다. 이 객체는 ifInOctets 의 64 비트 버전입니다.
ifHCInUcastPkts	1.3.6.1.2.1.31.1.1.1.7	이 서버 레이어에 의해 상위 (서브) 레이어에 전달되고 이 서버 레이어에서 멀티캐스트 또는 브로드캐스트 주소로 지정되지 않은 패킷의 개수입니다. 이 객체는 ifOutUcastPkts 의 64 비트 버전입니다.

MIB 객체 이름	OID	설명
ifHCInMulticastPkts	1.3.6.1.2.1.31.1.1.1.8	이 서브 레이어에 의해 상위 (서브) 레이어에 전달되고 이 서브 레이어에서 멀티캐스트 주소로 지정된 패킷의 개수입니다. MAC 레이어 프로토콜에 대해 이것은 그룹 및 기능 주소를 모두 포함합니다. 이 객체는 ifInMulticastPkts 의 64 비트 버전입니다.
ifHCInBroadcastPkts	1.3.6.1.2.1.31.1.1.1.9	이 서브 레이어에 의해 상위 (서브) 레이어에 전달되고 이 서브 레이어에서 브로드캐스트 주소로 지정된 패킷의 개수입니다. 이 객체는 ifInBroadcastPkts 의 64 비트 버전입니다.
ifHCOutOctets	1.3.6.1.2.1.31.1.1.1.10	프레이밍 문자들을 포함하여 인터페이스에서 전송된 옥텟의 총 개수입니다. 이 객체는 ifOutOctets 의 64 비트 버전입니다.
ifHCOutUcastPkts	1.3.6.1.2.1.31.1.1.1.11	폐기되었거나 송신되지 않은 것들을 포함하여 상위 레벨 프로토콜이 요청한 서브 네트워크 - 유니캐스트 주소로 전송되는 패킷의 총 개수입니다. 이 객체는 ifOutUcastPkts 의 64 비트 버전입니다.
ifHCOutMulticastPkts	1.3.6.1.2.1.31.1.1.1.12	폐기되었거나 송신되지 못한 것들을 포함하여 상위 레벨 프로토콜이 전송을 요청했고 이 서브 레이어에서 멀티캐스트 주소로 지정된 패킷의 총 개수입니다. MAC 레이어 프로토콜에 대해 이것은 그룹 및 기능 주소를 모두 포함합니다. 이 객체는 ifOutUcastPkts 의 64 비트 버전입니다.
ifHCOutBroadcastPkts	1.3.6.1.2.1.31.1.1.1.13	폐기되었거나 송신되지 않은 것들을 포함하여 상위 프로토콜이 전송을 요청했고 이 서브 레이어에서 브로드캐스트 주소로 지정된 패킷의 총 개수입니다. 이 객체는 ifOutBroadcastPkts 의 64 비트 버전입니다.

가

객체

폴링된 , **27**

ifEntry MIB 테이블 , **27**

MIB-II 확장 테이블에서 , **30**

구성

다중 폴러 , **20**

폴링 규칙 , **14**

규칙 기반 폴링 그룹 , **24**

다

다중 폴러 , **20**

대체 식별자 , **19**

데이터파이프

설치 , **10**

제거 , **11**

데이터파이프 설치 해제 , **11**

데이터파이프 업그레이드 , **10**

등록 정보 가져오기 폴링 플래그 설정 , **18**

마바

매뉴얼 , 목록 , **7**

벤더별 OVPI 유형 발견 파일 , **19**

분산 환경 설치 , **10**

사

색인 다시 달기 지원 , **19**

설치

분산 환경 , **10**

지시 , **10**

식별자 , 대체 , **19**

아

요구되는 소프트웨어 , **9**

유형 발견 파일 , **19**

인터페이스 식별자 , 변경 , **19**

자

전제 조건 , 소프트웨어 , **9**

제품 매뉴얼 검색 페이지 , **8**

제품 문서화 , **7**

지시된 - 인스턴스 폴링 , **25**

파

폴러 , 다중 , **20**

폴링 규칙

구성 , **14**

설명된 , **14**

폴링 그룹

규칙 기반 , **24**

기본값 , **24**

설명 , **23**

폴링 정책 관리자

새로운 폴러 생성 , **20**

폴링 플래그 설정 , **17**

폴링 정책 , 표준 , **24**

폴링 플래그 , **16**

등록 정보 가져오기로 설정 , **18**

폴링 정책 관리자로 설정 , **17**

폴링되는 객체 , 목록 , **27**

C

Cisco 라우터 , 대체 식별자 , **19**

Cisco Catalyst 스위치 , 대체 식별자 , **19**

G

GENMIB2IF_Datapipe 폴링 그룹 , 23

H

HP ProCurve 스위치 , 대체 식별자 , 19

I

ifAdminStatus 객체 설명 , 28

ifAlias 객체 설명 , 30

ifDescr 객체 설명 , 27

ifEntry MIB 테이블 , 객체 , 27

ifHCInBroadcastPkts 객체 설명 , 31

ifHCInMulticastPkts 객체 설명 , 31

ifHCInOctets 객체 설명 , 30

ifHCInUcastPkts 객체 설명 , 30

ifHCOutBroadcastPkts 객체 설명 , 31

ifHCOutMulticastPkts 객체 설명 , 31

ifHCOutOctets 객체 설명 , 31

ifHCOutUcastPkts 객체 설명 , 31

ifIndex 객체 설명 , 27

ifInDiscards 객체 설명 , 28

ifInErrors 객체 설명 , 28

ifInNUcastPkts 객체 설명 , 28

IfInOctets 객체 설명 , 28

ifInUcastPkts 객체 설명 , 28

ifLastChange 객체 설명 , 28

ifMtu 객체 설명 , 27

ifName 객체 설명 , 30

ifOperStatus 객체 설명 , 28

ifOutDiscards 객체 설명 , 29

ifOutErrors 객체 설명 , 29

ifOutNUcastPkts 객체 설명 , 29

ifOutOctets 객체 설명 , 28

ifOutQLen 객체 설명 , 29

ifOutUcastPkts 객체 설명 , 28

ifPhysAddress 객체 설명 , 28

ifSpecific 객체 설명 , 29

ifSpeed 객체 설명 , 27

ifType 객체 설명 , 27

ifUnknownProtos 객체 설명 , 28

index-by-ifAlias 폴링 그룹 , 23

index-by-ifDescr 폴링 그룹 , 23

index-by-ifIndex 폴링 그룹 , 23, 24

index-by-ifName 폴링 그룹 , 23

Interface Reporting 등록 정보 가져오기
폴링 플래그 설정 , 18

IR_map_p 저장 절차 , 25

IRifEntry Sync Interfaces 패키지 , 18

K

K_IFEntry_Disc 등록 정보 테이블 , 25

M

MIB 객체 , 목록 , 27

MIB-II 확장 테이블 , 객체 , 30

O

OVPI 타이머 , 11

P

poll_flag_ifEntry, 25

ProCurve 스위치 , 대체 식별자 , 19

S

SQL 스크립트 , 폴링 플래그 설정의 , 19